Higher-order Statistical Modeling based Deep CNNs （Part-II）

Global Second-order Pooling for Deep CNN

Li Peihua, Wang Qilong

2018-11-23

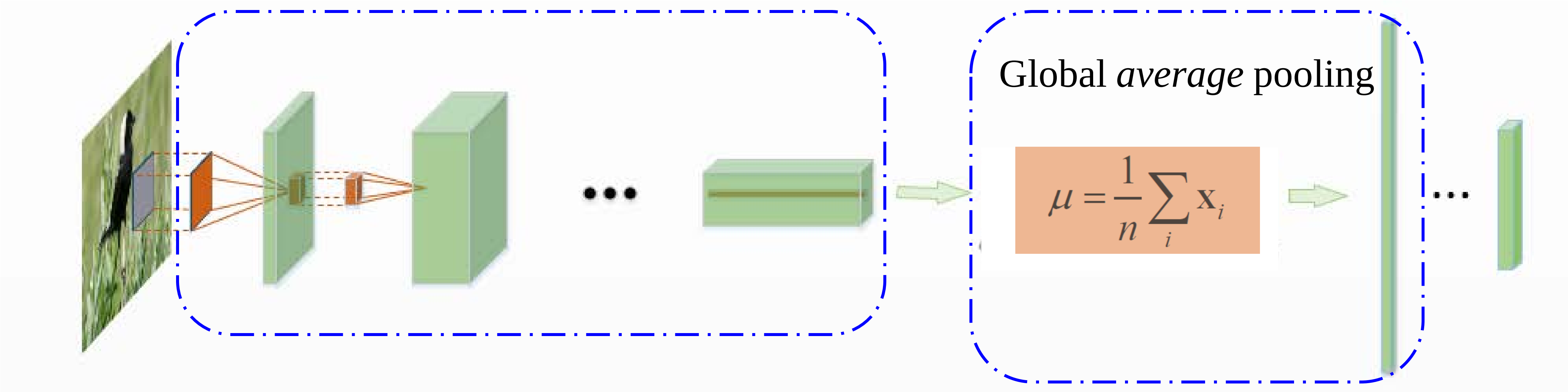
Outline

- **Global Second-order Pooling**
- **Matrix Power Normalizatin and Fast Training**
- **Global Distribution Modeling for CNN**
- **Conclusion**

Global Second-order Pooling

- Deep CNN

Learning Represenation from Local to Global



Image

Local Representation of *image regions* of increasing receptive field

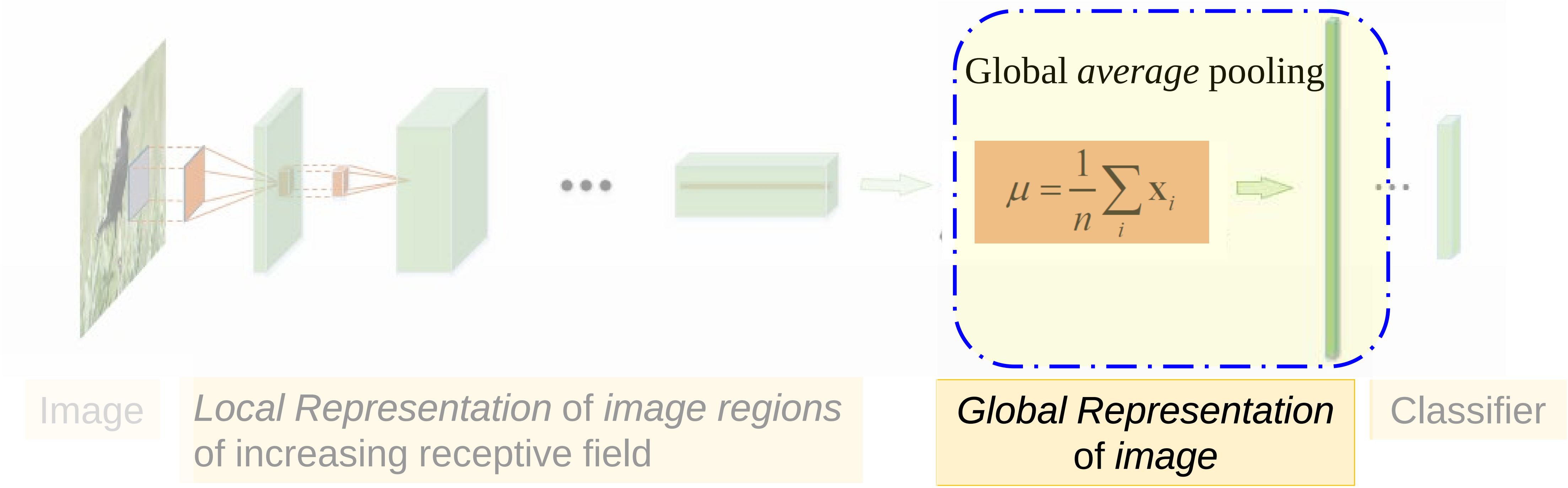
Global Representation of *image*

Classifier

Global Second-order Pooling

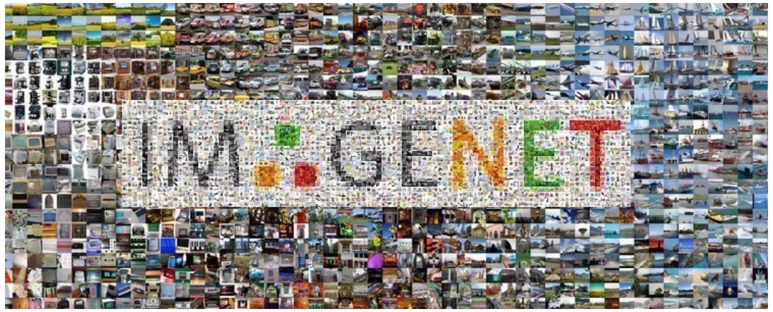
- Global Average Pooling:

  Widespread in Inception, ResNet, DenseNet etc.



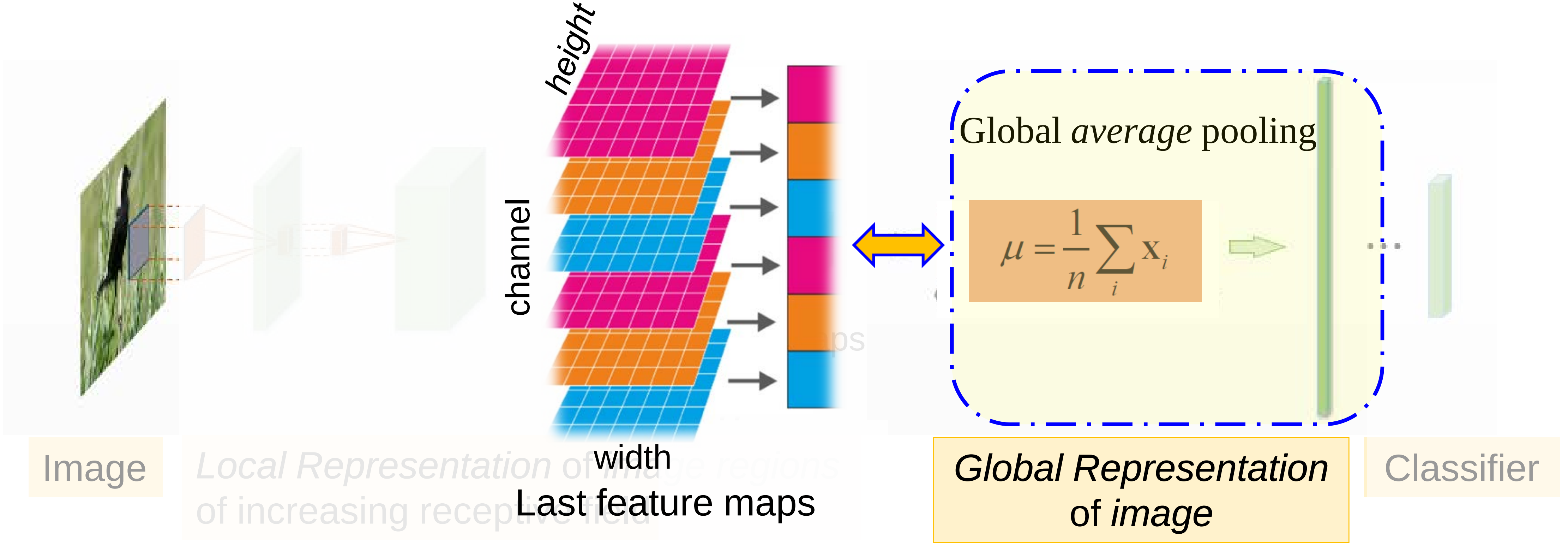
Min Lin, Qiang Chen, Shuicheng Yan. Network in network. In ICLR, 2014.

Global Second-order Pooling



- Global Average Pooling:

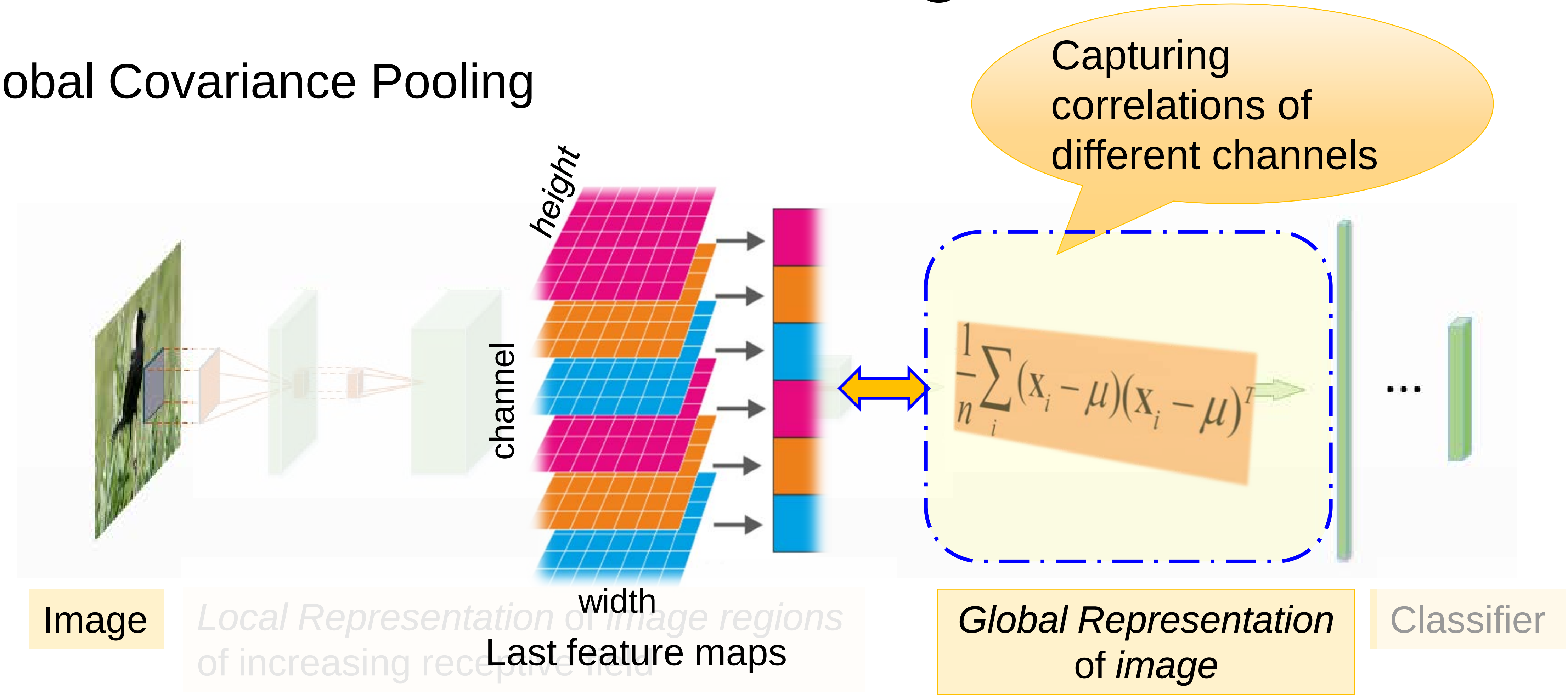
  Widespread in Inception, ResNet, DenseNet etc.



Min Lin, Qiang Chen, Shuicheng Yan. Network in network. In ICLR, 2014.

Global Second-order Pooling

- Global Covariance Pooling



Global Second-order Pooling

From **O_2P (ECCV'12**) to DeepO_2P (ICCV'15)

J. Carreira et al. Semantic Segmentation with Second-Order Pooling. in *ECCV*, 2012.
J. Carreira *et al.* Freeform region description with second-order pooling. *IEEE TPAMI*, 2015.

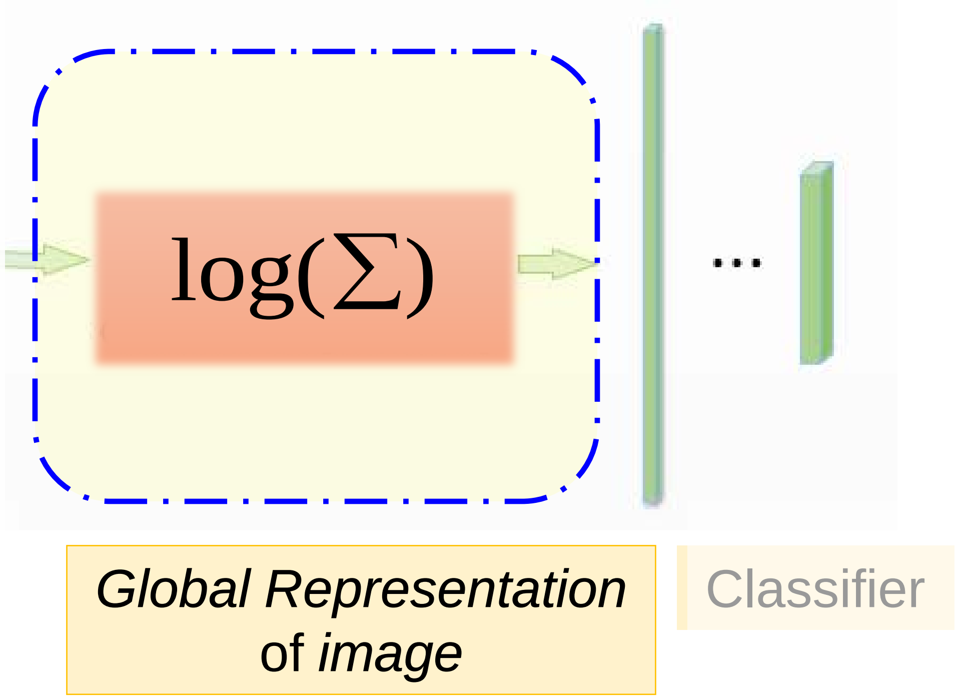
Global Second-order Pooling

- From O_2P (ECCV'12) to **$\mathbf{DeepO_2P}$ (ICCV'15)**

➢First end-to-end global covariance pooling

➢Backprogation of Singlular Value Decompositon (SVD)

➢Performance is not competitive

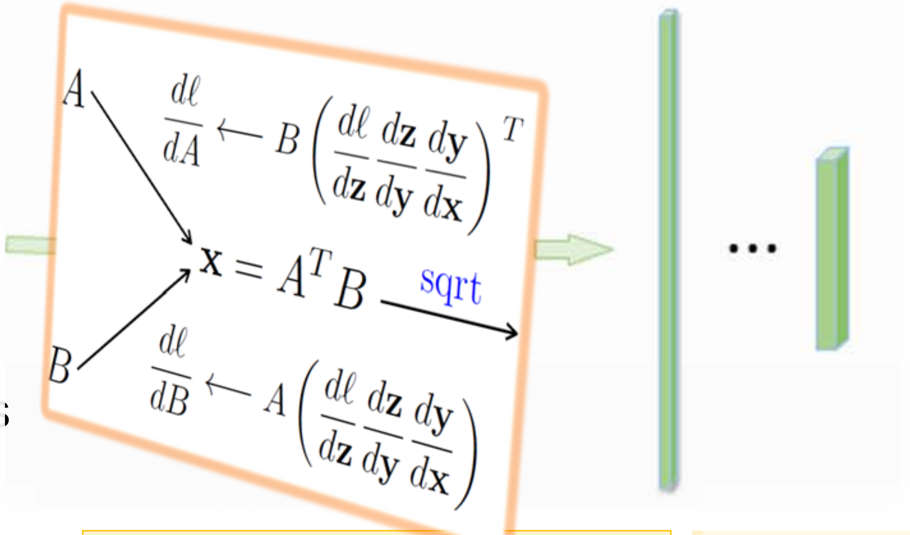


[$DeepO_2P$] C. Ionescu *et al.* **Matrix Backpropagation for Deep Networks with Structured Layers. In ICCV, 2015.**

Global Second-order Pooling

- Global Covariance Pooling — **Blinear CNN (i.e. B-CNN, ICCV'15)**

  - ➢First end-to-end global covariance pooling
  - ➢Element-wise SQRT+l_2 normalziation
  - ➢Competitive performance on Fine-grained classification



[B-CNN] T.-Y. Lin, A. RoyChowdhury, S. Maji. Bilinear CNN models for fine-grained visual recognition. In ICCV, 2015.

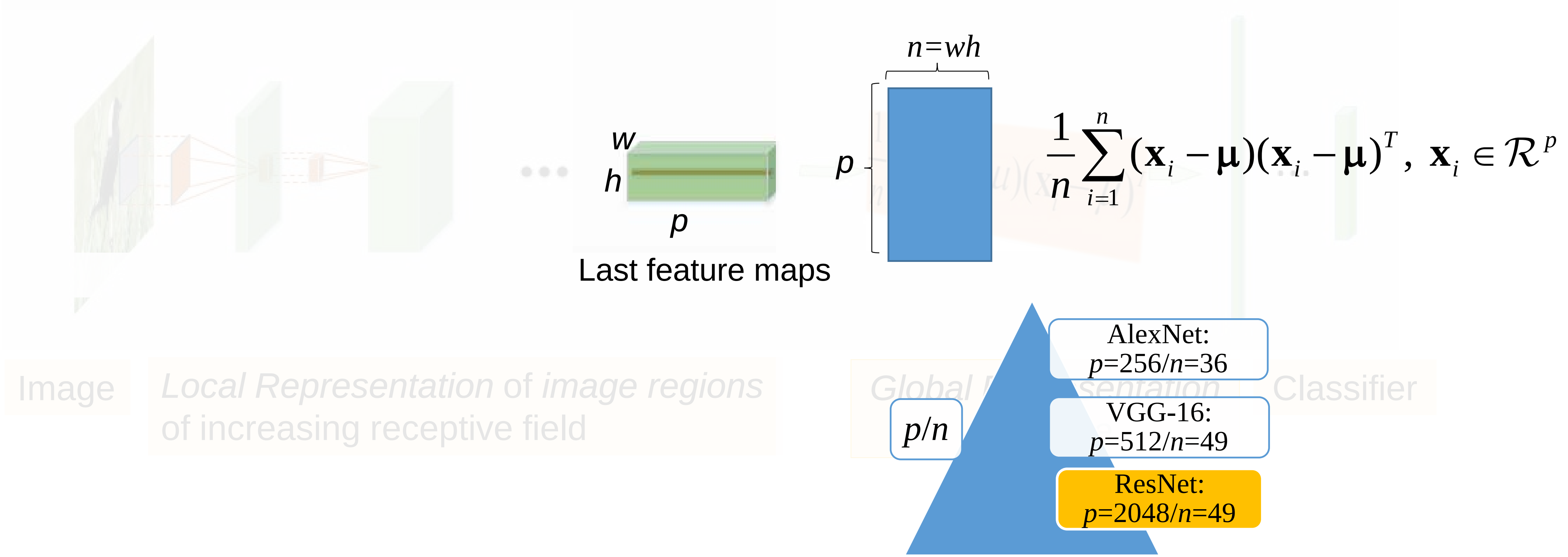
Global Second-order Pooling

Will global 2^{nd}-order pooling work on

DeepO$_2$P and B-CNN fail to perform well

Global Second-order Pooling

- **Challenges** of Global Covariance Pooling in CNN
  1. Statistical problem of small sample/high-dimensionality(n<p)

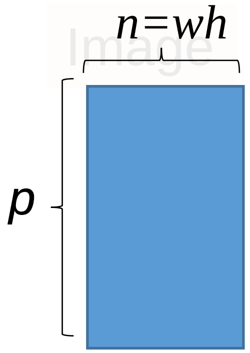


Global Second-order Pooling

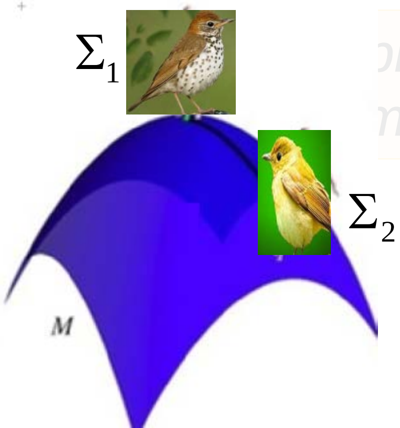
- **Challenges** of Global Covariance Pooling in CNN

1. Statistical problem of Small sample/high-dimensionality(n<p)

2. Manifold structure of covariance matrices



$$\Sigma = \frac{1}{n}\sum_{i=1}^{n}(\mathbf{x}_i - \boldsymbol{\mu})(\mathbf{x}_i - \boldsymbol{\mu})^T$$



Geodesic distance needs to be considered

Global Second-order Pooling

DeepO$_2$P and B-CNN fail to well address :

1. Statistical small sample/high dimensionality;
2. Manifold structure of covariance matrices;
3. Whether work on large-scale IMAGENET .

[DeepO$_2$P] C. Ionescu *et al.* Matrix Backpropagation for Deep Networks with Structured Layers. In ICCV, 2015.

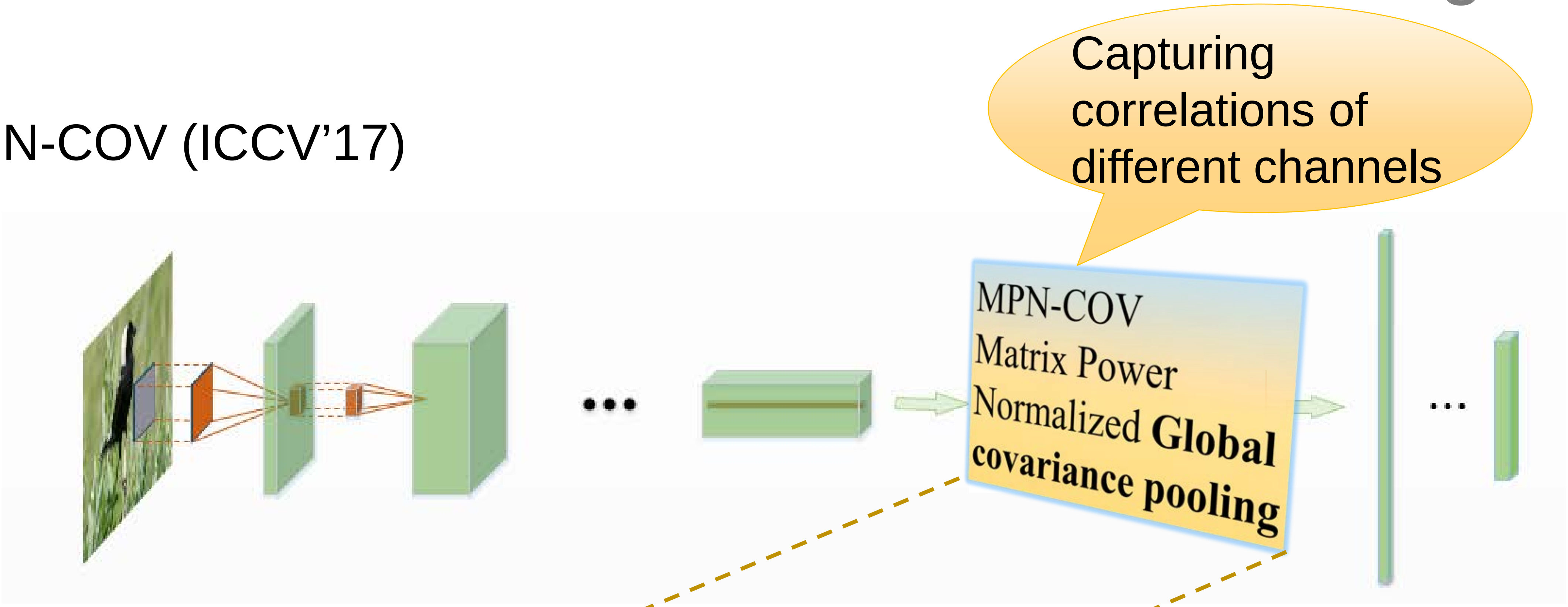
[B-CNN] T.-Y. Lin, A. RoyChowdhury, S. Maji. Bilinear CNN models for fine-grained visual recognition. In ICCV, 2015.

Outline

- Global Covariance Pooling
- **Matrix Power Normalizatin and Fast Training**
- Global Distribution Modeling for CNN
- Conclusion

Matrix Power Normalization and Fast Training

- MPN-COV (ICCV'17)



$\alpha = 0.5$ performing best

$$\left(\frac{1}{n} \sum_i (\mathrm{x}_i - \mu)(\mathrm{x}_i - \mu)^T \right)^\alpha$$

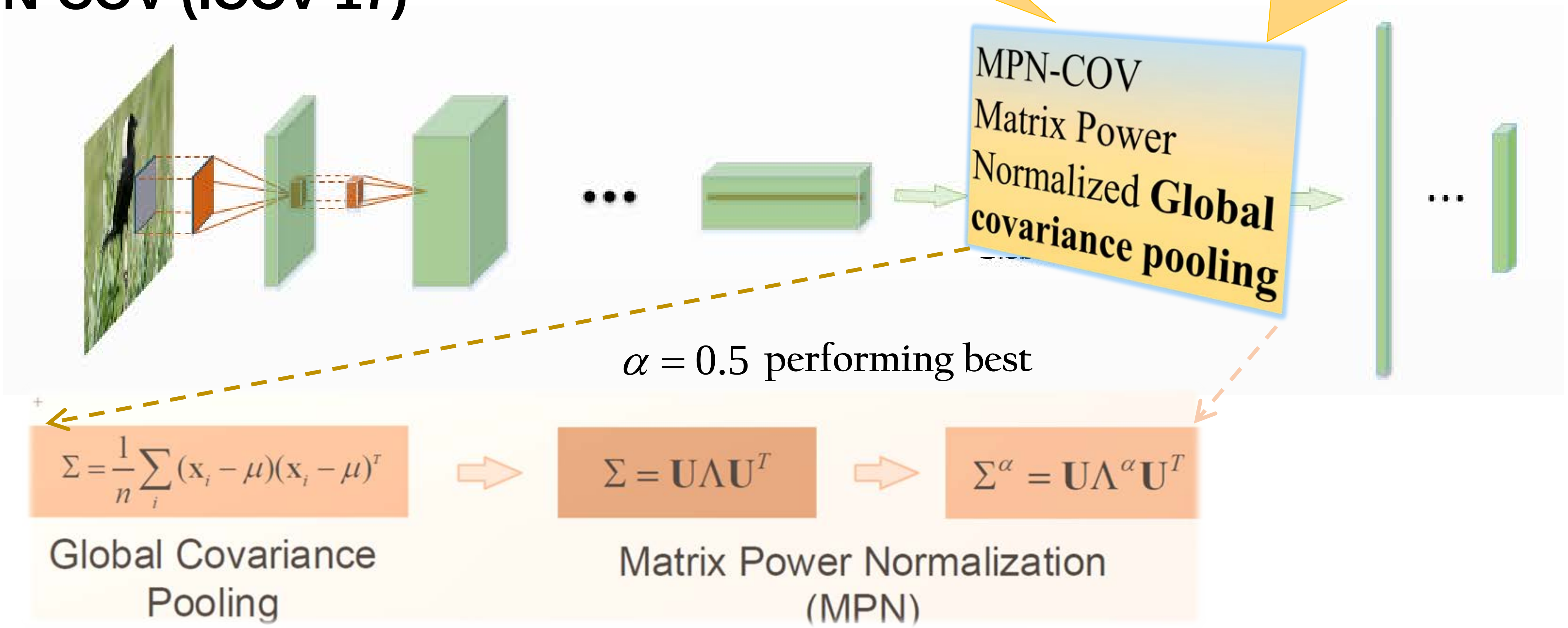
[MPN-COV] Peihua Li, Jiangtao Xie, Qilong Wang and Wangmeng Zuo. Is Second-order Information Helpful for Large-scale Visual Recognition? In ICCV, 2017.

Matrix Power Normalization and Fast Training

Challenge 1
Small sample/high-dimensionality, e.g. ResNet, 49/2048

Challenge 2
Exploiting geometry of covariance spaces

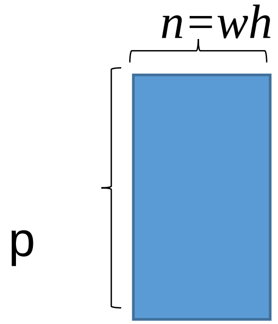
- **MPN-COV (ICCV'17)**



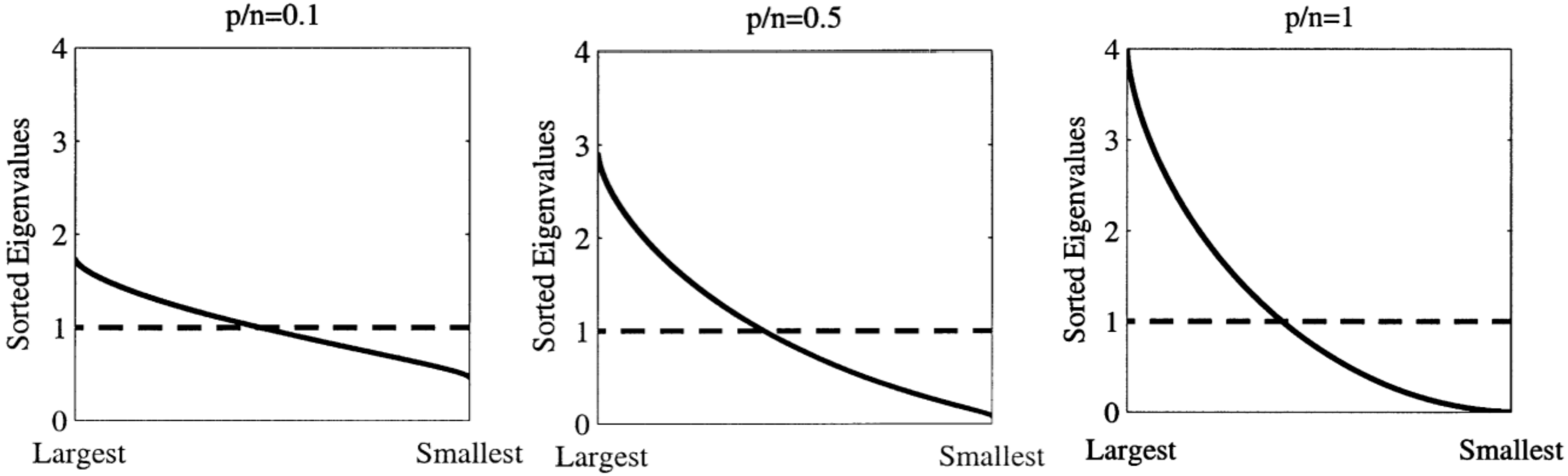
[MPN-COV] Peihua Li, Jiangtao Xie, Qilong Wang and Wangmeng Zuo. Is Second-order Information Helpful for Large-scale Visual Recognition? In ICCV, 2017.

Why MPN-COV works: **Statistical Insight (qualitatively)**

Small sample/high-dimensionality (n < p)



$$\frac{1}{n}\sum_{i=1}^{n}(\mathbf{x}_i-\boldsymbol{\mu})(\mathbf{x}_i-\boldsymbol{\mu})^T, \mathbf{x}_i \in \mathcal{R}^p$$



Adverse effect: with p/n ↑, largest (resp. smallest) values tends to be much larger (smaller) than ground truths

O. Ledoit and M. Wolf. A well-conditioned estimator for large-dimensional covariance matrices. J. Multivariate Analysis, 88(2):365–411, 2004.

C. Stein. Lectures on the theory of estimation of many parameters. Journal of Soviet Mathematics, 34(1):1373–1403, 1986.

Why MPN-COV works: **Statistical Insight (qualitatively-FP**

$$\mathbf{U}\begin{bmatrix} \log(\lambda_1) & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \log(\lambda_p) \end{bmatrix}\mathbf{U}^T$$

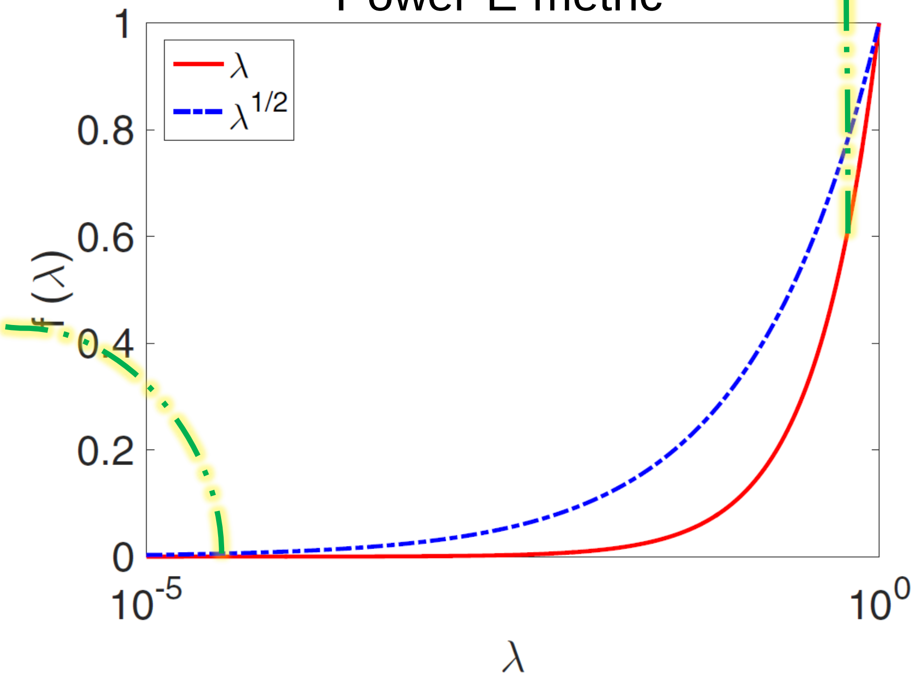
$\log(\lambda)$

log-E metric

$$\mathbf{U}\begin{bmatrix} \boldsymbol{\lambda}_1^{\alpha} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \boldsymbol{\lambda}_p^{\alpha} \end{bmatrix}\mathbf{U}^T \quad 0.64 \rightarrow 0.8$$

Power-E metric

$10^{-4} \rightarrow 10^{-2}$



$\lambda^{1/2}$

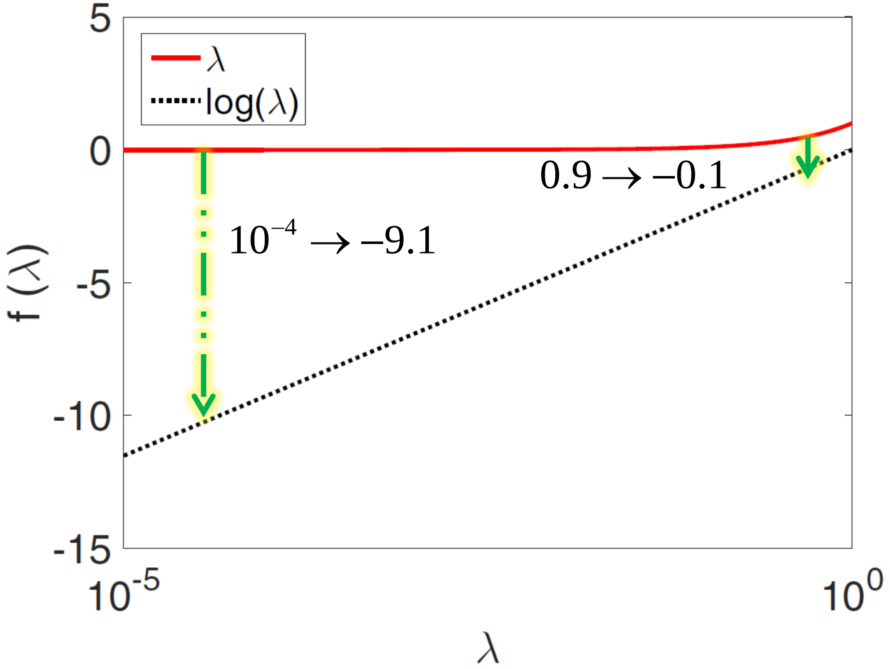
Power-E metric relatively shrinks largest eigenvalues while stretchs smallest eigenvalues

Why MPN-COV works: **Statistical Insight (qualitatively-FP)**

$\log(\lambda)$

log-E metric

$$\mathbf{U}\begin{bmatrix} \log(\lambda_1) & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \log(\lambda_p) \end{bmatrix}\mathbf{U}^T$$



Log-E metric over stretchs smallest eigenvalues, changing order of significance of eigenvalues

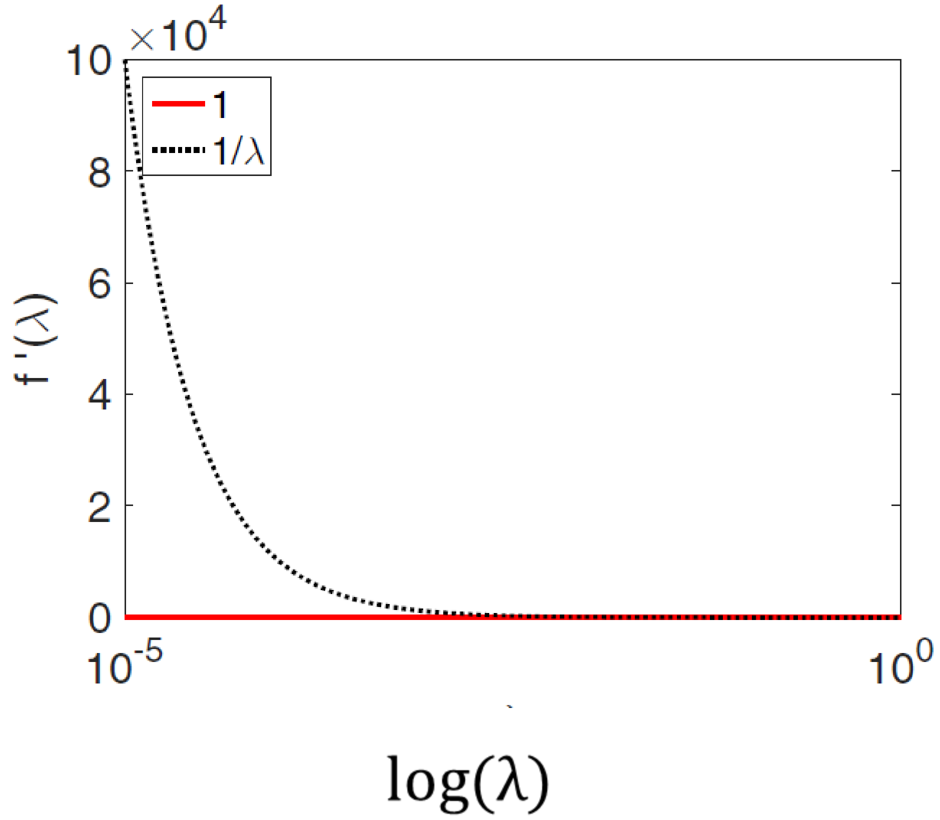
$$\mathbf{U}\begin{bmatrix} \lambda_1^{\alpha} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \lambda_p^{\alpha} \end{bmatrix}\mathbf{U}^T$$

$\lambda^{1/2}$

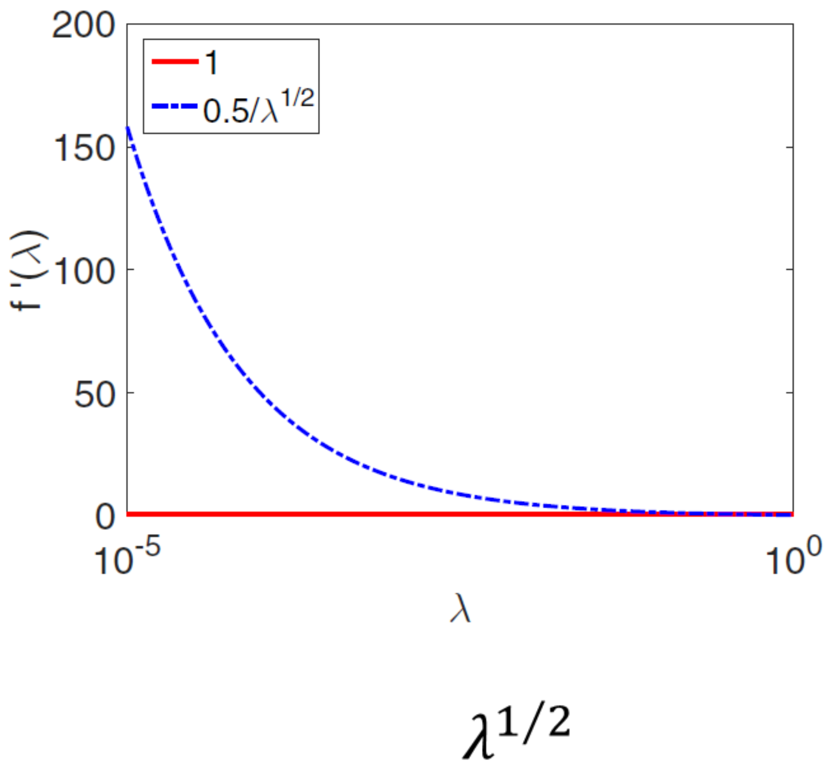
Power-E metric

Why MPN-COV works: **Statistical Insight (qualitatively-FP)**

- Log-E: Smallest eigenvalues affect the **gradient** considerably



log-E metric



Power-E metric

Why MPN-COV works: **Statistical Insight (quantitatively)**

Classical MLE

$$\underbrace{\frac{1}{N}\sum_{i=1}^{N}(\mathbf{x}_i-\boldsymbol{\mu})(\mathbf{x}_i-\boldsymbol{\mu})^T}_{\mathbf{P}} = \arg\min_{\boldsymbol{\Sigma}} \log|\boldsymbol{\Sigma}| + \mathrm{tr}(\boldsymbol{\Sigma}^{-1}\mathbf{P})$$

[RAID-G] Qilong Wang, Peihua Li, Wangmeng Zuo, Lei Zhang. RAID-G: Robust Estimation of Approximate Infinite Dimensional Gaussian with Application to Materiel Recognition. In CVPR, 2016

Why MPN-COV works: **Statistical Insight (quantitatively)**

- Regularized Maximum Likelihood Estimation (MLE) Method: vN-MLE

Classical MLE $\Rightarrow$ $$\underbrace{\frac{1}{N}\sum_{i=1}^{N}(\mathbf{x}_i-\boldsymbol{\mu})(\mathbf{x}_i-\boldsymbol{\mu})^T}_{\mathbf{P}} = \arg\min_{\boldsymbol{\Sigma}} \log|\boldsymbol{\Sigma}| + \mathrm{tr}(\boldsymbol{\Sigma}^{-1}\mathbf{P})$$

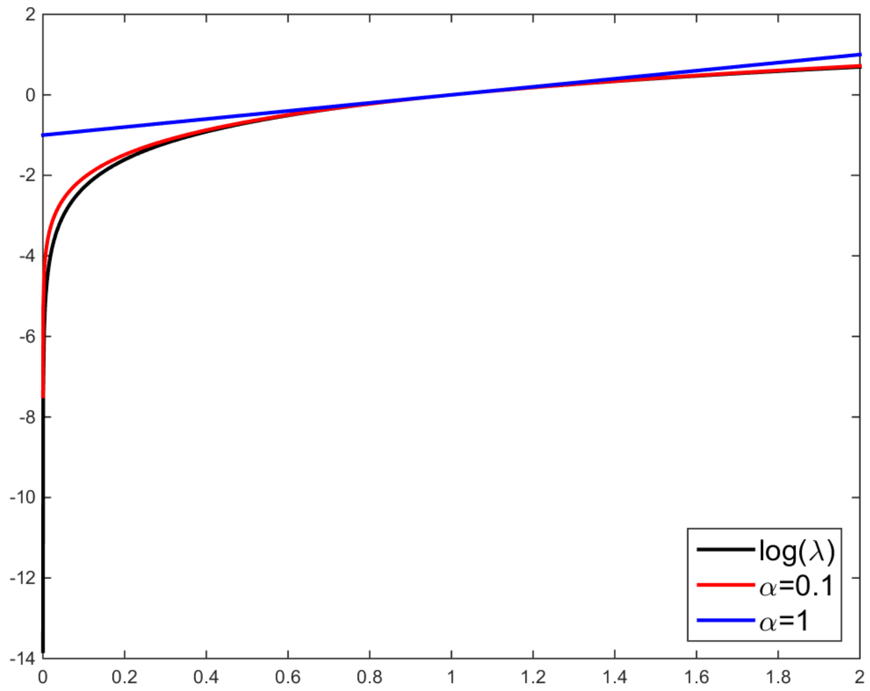
MPN-COV with $\alpha = 1/2$ is the unique solution to the regularized MLE of covariance matrix, i.e.,

$$\mathbf{P}^{\frac{1}{2}} = \arg\min_{\boldsymbol{\Sigma}} \log|\boldsymbol{\Sigma}| + \mathrm{tr}\left(\boldsymbol{\Sigma}^{-1}\mathbf{P}\right) + D_{vN}\left(\mathbf{I},\boldsymbol{\Sigma}\right).$$

[RAID-G] Qilong Wang, Peihua Li, Wangmeng Zuo, Lei Zhang. RAID-G: Robust Estimation of Approximate Infinite Dimensional Gaussian with Application to Materiel Recognition. In CVPR, 2016

Why MPN-COV works—**Geometric Insight**

- Power Euclidean $d_\alpha(\mathbf{P}, \widetilde{\mathbf{P}}) = \frac{1}{\alpha}\left\|\mathbf{P}^\alpha - \widetilde{\mathbf{P}}^\alpha\right\|_F$ exploits gemotry

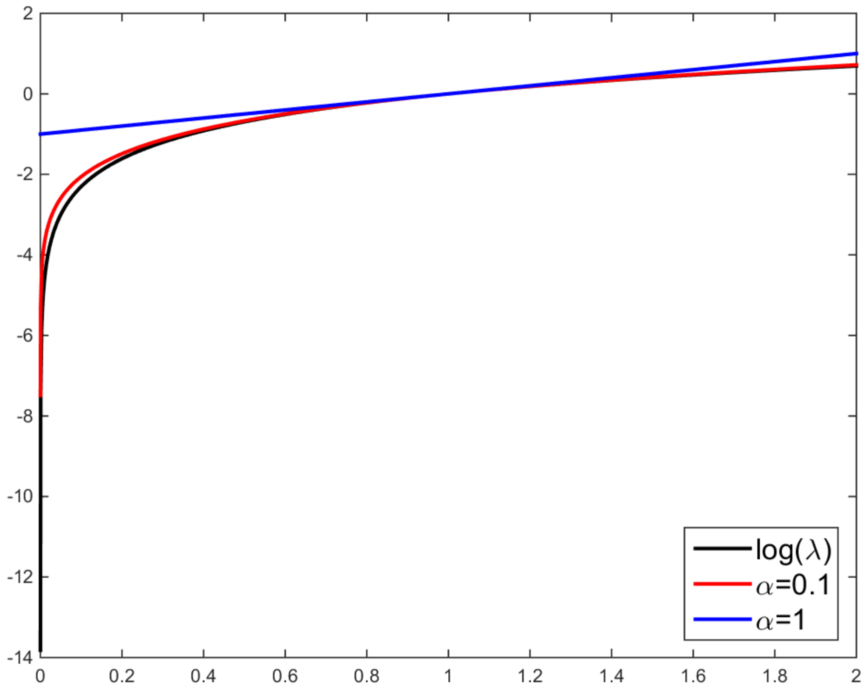


$$\lim_{\alpha\to 0} d_\alpha(\mathbf{P}, \widetilde{\mathbf{P}}) = \underbrace{\left\| \log(\mathbf{P}) - \log(\widetilde{\mathbf{P}}) \right\|_F}_{\text{log-E metric}}$$

[MPN-COV] Peihua Li, Jiangtao Xie, Qilong Wang and Wangmeng Zuo. Is Second-order Information Helpful for Large-scale Visual Recognition? In ICCV, 2017.

Why MPN-COV works—**Geometric Insight**

- Exploiting geometry of covariance spaces $d_{\alpha}(\mathbf{P},\widetilde{\mathbf{P}}) = \frac{1}{\alpha}\left\|\mathbf{P}^{\alpha} - \widetilde{\mathbf{P}}^{\alpha}\right\|_F$

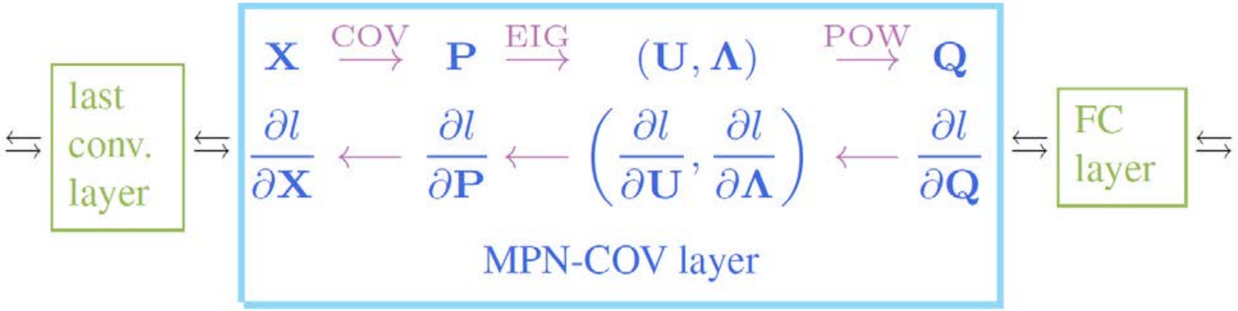


$$\lim_{\alpha\to 0} d_{\alpha}(\mathbf{P},\widetilde{\mathbf{P}}) = \underbrace{\left\|\log(\mathbf{P}) - \log(\widetilde{\mathbf{P}})\right\|_F}_{\text{log-E metric}}$$

Log-E metric	Power-E metric
measures true geodesic distance	measures it approximately
The input must be strictly positive	allows non-negative number

[MPN-COV] Peihua Li, Jiangtao Xie, Qilong Wang and Wangmeng Zuo. Is Second-order Information Helpful for Large-scale Visual Recognition? In ICCV, 2017.

Matrix Power Normalization and Fast Training



Forward propagation

$$\mathbf{X} \mapsto \mathbf{P}, \quad \mathbf{P} = \mathbf{X}\bar{\mathbf{I}}\mathbf{X}^T$$

$$\mathbf{P} \mapsto (\mathbf{U}, \Lambda), \quad \mathbf{P} = \mathbf{U}\Lambda\mathbf{U}^T$$

$$(\mathbf{U}, \Lambda) \mapsto \mathbf{Q}, \quad \mathbf{Q} = \mathbf{U}\mathbf{F}(\Lambda)\mathbf{U}^T$$

Backward propagation

$$\mathrm{tr}\left(\left(\frac{\partial l}{\partial \mathbf{X}}\right)^T \mathrm{d}\,\mathbf{X}\right) = \mathrm{tr}\left(\left(\frac{\partial l}{\partial \mathbf{P}}\right)^T \mathrm{d}\,\mathbf{P}\right)$$

$$\mathrm{tr}\left(\left(\frac{\partial l}{\partial \mathbf{P}}\right)^T \mathrm{d}\,\mathbf{P}\right) = \mathrm{tr}\left(\left(\frac{\partial l}{\partial \mathbf{U}}\right)^T \mathrm{d}\,\mathbf{U} + \left(\frac{\partial l}{\partial \Lambda}\right)^T \mathrm{d}\,\Lambda\right)$$

$$\mathrm{tr}\left(\left(\frac{\partial l}{\partial \mathbf{U}}\right)^T \mathrm{d}\,\mathbf{U} + \left(\frac{\partial l}{\partial \Lambda}\right)^T \mathrm{d}\,\Lambda\right) = \mathrm{tr}\left(\left(\frac{\partial l}{\partial \mathbf{Q}}\right)^T \mathrm{d}\,\mathbf{Q}\right)$$

[MPN-COV] Peihua Li, Jiangtao Xie, Qilong Wang and Wangmeng Zuo. Is Second-order Information Helpful for Large-scale Visual Recognition? In ICCV, 2017.

Matrix Power Normalization and Fast Training

$$\frac{\partial l}{\partial \mathbf{U}} = \left(\frac{\partial l}{\partial \mathbf{Q}} + \left(\frac{\partial l}{\partial \mathbf{Q}}\right)^T\right)\mathbf{U}\mathbf{F}$$

$$\frac{\partial l}{\partial \boldsymbol{\Lambda}} \begin{cases} \alpha\left(\mathrm{diag}\left(\lambda_1^{\alpha-1},\ldots,\lambda_d^{\alpha-1}\right)\mathbf{U}^T\frac{\partial l}{\partial \mathbf{Q}}\mathbf{U}\right)_{\mathrm{diag}} & \text{MPN} \\ \frac{\alpha}{\lambda_1^{\alpha}}\left(\mathrm{diag}\left(\lambda_1^{\alpha-1},\ldots,\lambda_d^{\alpha-1}\right)\mathbf{U}^T\frac{\partial l}{\partial \mathbf{Q}}\mathbf{U}\right)_{\mathrm{diag}} - \mathrm{diag}\left(\frac{\alpha}{\lambda_1}\mathrm{tr}\left(\mathbf{Q}\frac{\partial l}{\partial \mathbf{Q}}\right),0,\ldots,0\right) & \text{MPN} + \text{M-}l_2 \\ \ldots & \end{cases}$$

$$\frac{\partial l}{\partial \mathbf{P}} = \mathbf{U}\left(\left(\mathbf{K}^T \circ \left(\mathbf{U}^T\frac{\partial l}{\partial \mathbf{U}}\right)\right) + \left(\frac{\partial l}{\partial \boldsymbol{\Lambda}}\right)_{\mathrm{diag}}\right)\mathbf{U}^T$$

$$\frac{\partial l}{\partial \mathbf{X}} = \bar{\mathbf{I}}\mathbf{X}\left(\frac{\partial l}{\partial \mathbf{P}} + \left(\frac{\partial l}{\partial \mathbf{P}}\right)^T\right)$$

[MPN-COV] Peihua Li, Jiangtao Xie, Qilong Wang and Wangmeng Zuo. Is Second-order Information Helpful for Large-scale Visual Recognition? In ICCV, 2017.

Matrix Power Normalization and Fast Training

- Downside of Eigendecomposition (or SVD)

$$\Sigma^{\frac{1}{2}}$$

MPN-COV (ICCV17)

[MPN-COV] Peihua Li, Jiangtao Xie, Qilong Wang and Wangmeng Zuo. Is Second-order Information Helpful for Large-scale Visual Recognition? In ICCV, 2017.
[G²DeNet] Qilong Wang, Peihua Li, Lei Zhang. G²DeNet: Global Gaussian Distribution Embedding Network and Its Application to Visual Recognition. In *CVPR*, 2017 (*Oral*).

Matrix Power Normalization and Fast Training

- Downside of Eigendecomposition (or SVD)

Inefficient due to GPU unfriendly EIG/SVD

$$\Sigma^{\frac{1}{2}} \qquad \Sigma = \mathbf{U}\Lambda\mathbf{U}^T \Rightarrow \Sigma^{\frac{1}{2}} = \mathbf{U}\Lambda^{\frac{1}{2}}\mathbf{U}^T$$

MPN-COV (ICCV17)

Time (ms) taken by EIG/SVD of 256x256 covariance matrix

Algorithm	CUDA cuSOLVER	Matlab (CPU function)	Matlab (GPU function)
EIG	21.3	1.8	9.8
SVD	52.2	4.1	11.9

[MPN-COV] Peihua Li, Jiangtao Xie, Qilong Wang and Wangmeng Zuo. Is Second-order Information Helpful for Large-scale Visual Recognition? In ICCV, 2017.

Matrix Power Normalization and **Fast Training**

- Iterative matrix square root (CVPR'18)

$$\Sigma = \frac{1}{n}\sum_i (\mathbf{x}_i - \mu)(\mathbf{x}_i - \mu)^T \Rightarrow \mathbf{\Sigma = U\Lambda U}^T \Rightarrow \mathbf{\Sigma^{\frac{1}{2}} = U\Lambda^{\frac{1}{2}}U}^T$$

Global Covariance Pooling

Matrix Power Normalization (MPN)

inefficient

Proposed method: iSQRT-COV

$$\Sigma = \frac{1}{n}\sum_i (\mathbf{x}_i - \mu)(\mathbf{x}_i - \mu)^T \Rightarrow \Sigma^{\frac{1}{2}} = \mathrm{iSQRT}(\Sigma)$$

Global Covariance Pooling

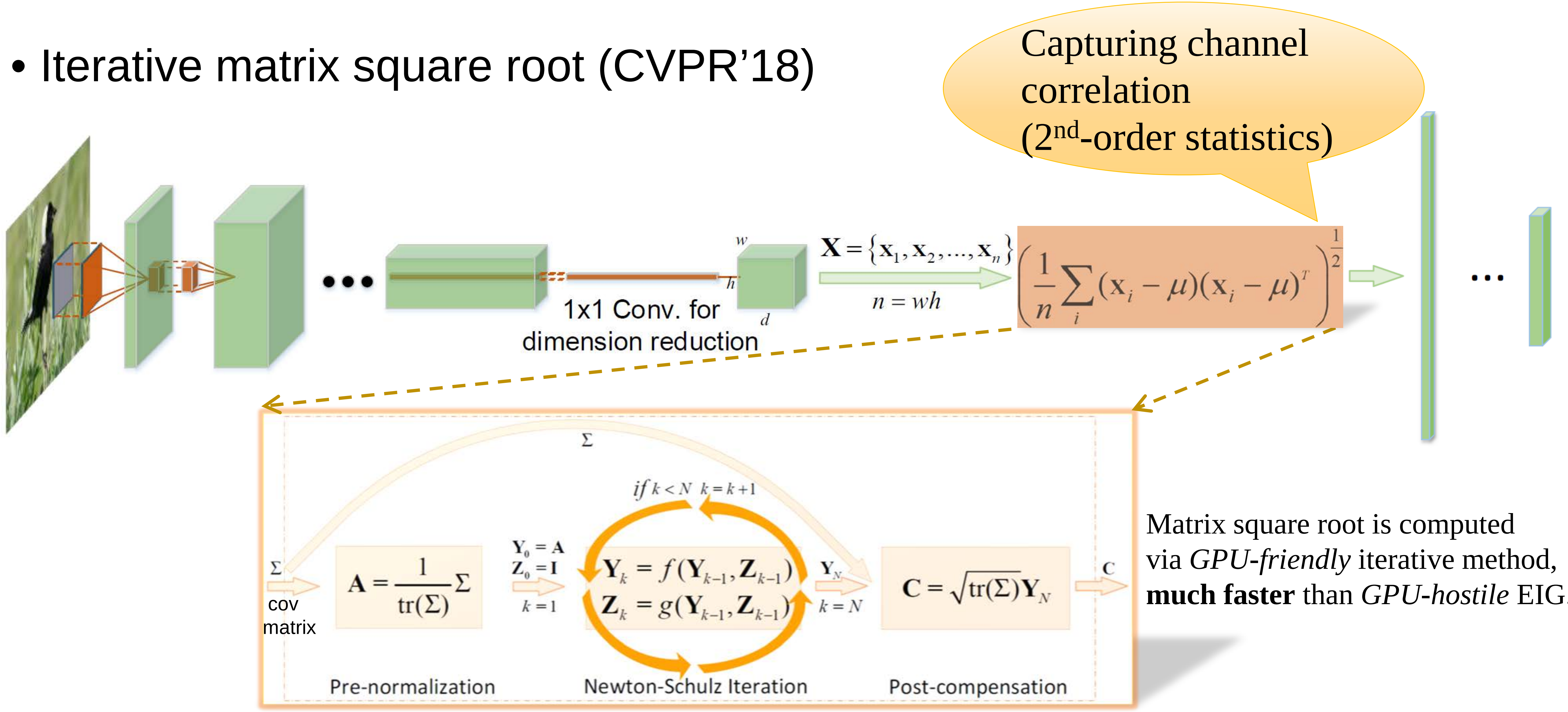
Iterative Matrix Square Root Normalization (iSQRT)

very fast!

[iSQRT-COV] Peihua Li, Jiangtao Xie, Qilong Wang and Zilin Gao. Towards Faster Training of Global Covariance Pooling Networks by Iterative Matrix Square Root Normalization. In *CVPR*, 2018.

Matrix Power Normalization and Fast Training

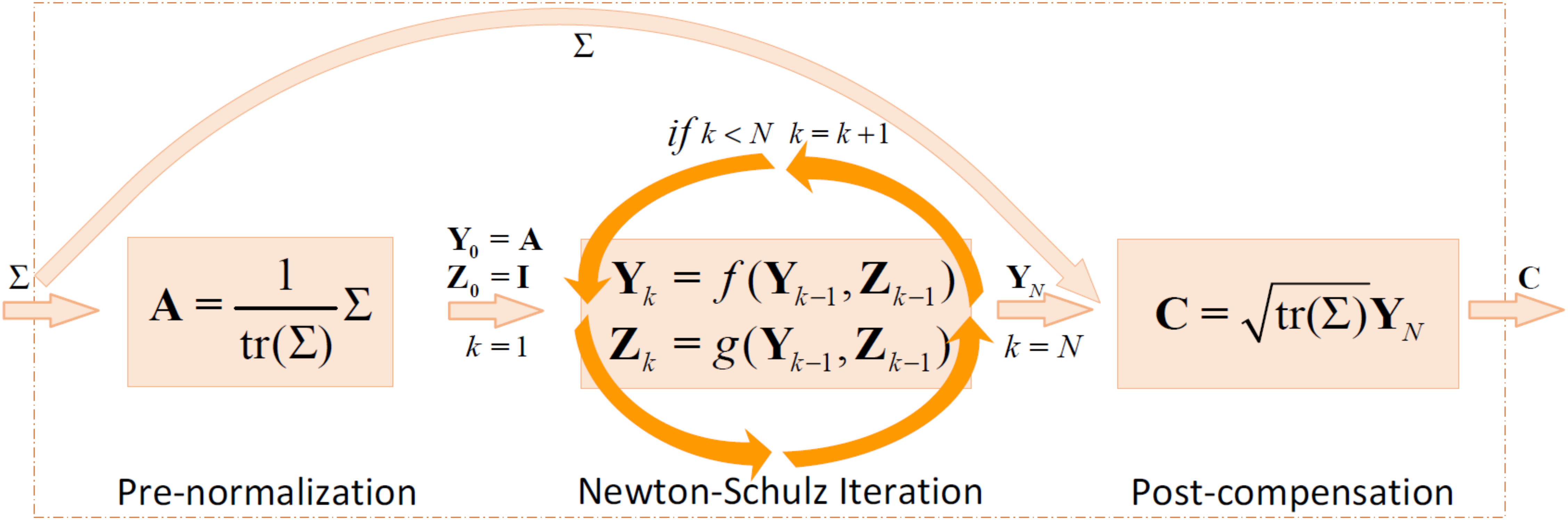
- Iterative matrix square root (CVPR'18)



Matrix square root is computed via *GPU-friendly* iterative method, **much faster** than *GPU-hostile* EIG

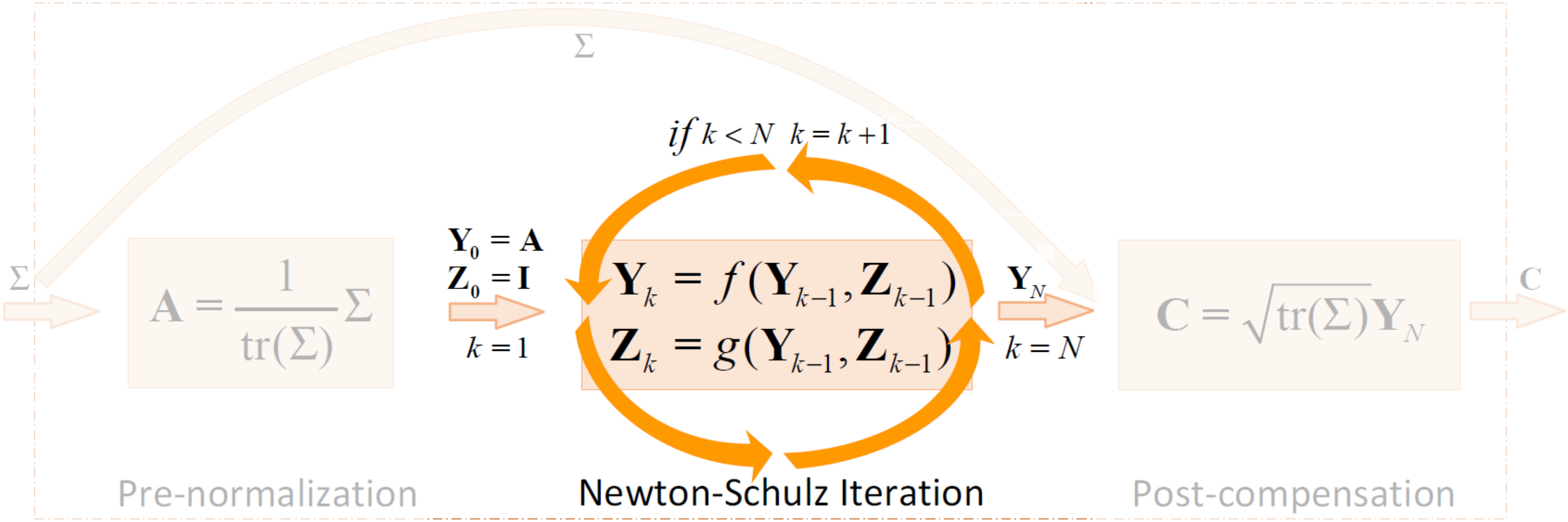
Iterative Matrix Square Root Normalization (CVPR'18)

A Directed Acyclic Graph (DAG) with Iteration



[iSQRT-COV] Peihua Li, Jiangtao Xie, Qilong Wang and Zilin Gao. Towards Faster Training of Global Covariance Pooling Networks by Iterative Matrix Square Root Normalization. In *CVPR*, 2018.

Iterative Matrix Square Root Normalization (CVPR'18)



Forward propagation

$$\mathbf{Y}_k = \frac{1}{2}\mathbf{Y}_{k-1}(3\mathbf{I} - \mathbf{Z}_{k-1}\mathbf{Y}_{k-1})$$

$$\mathbf{Z}_k = \frac{1}{2}(3\mathbf{I} - \mathbf{Z}_{k-1}\mathbf{Y}_{k-1})\mathbf{Z}_{k-1}$$

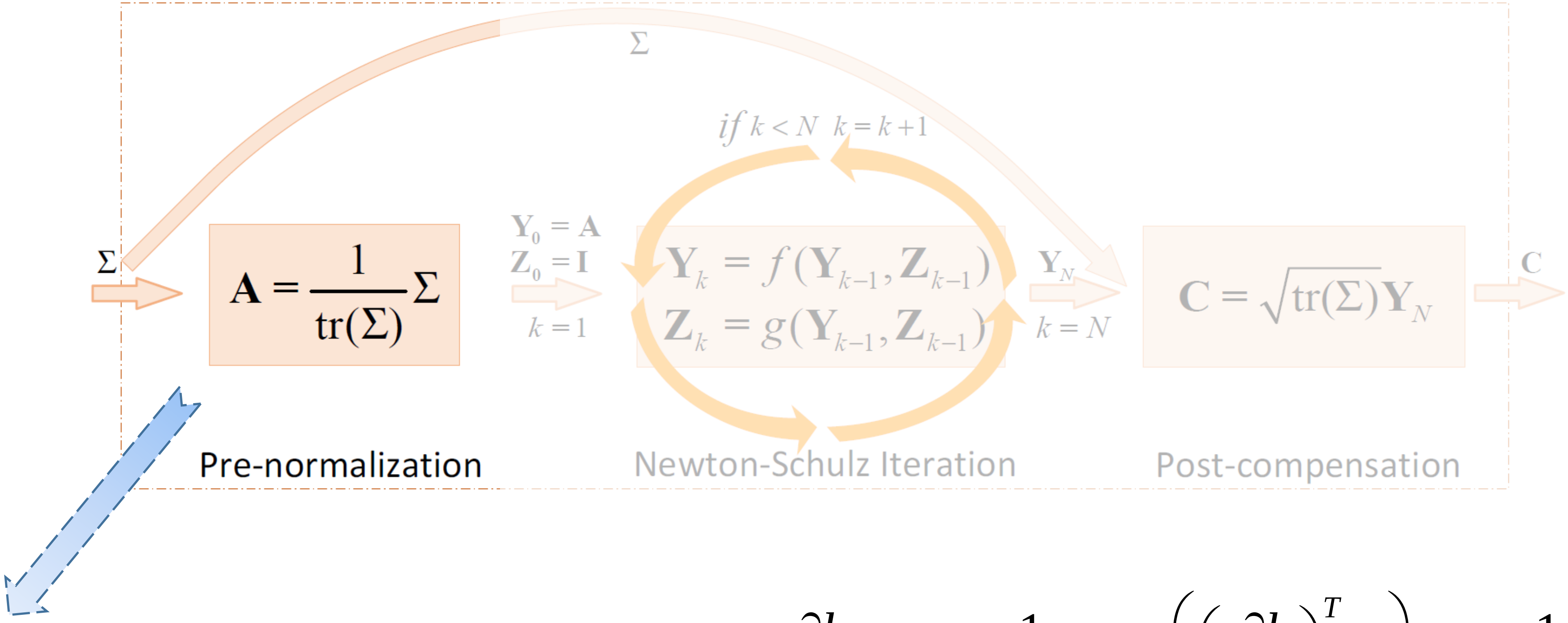
Backward gradient

$$\frac{\partial l}{\partial \mathbf{Y}_{k-1}} = \frac{1}{2}\left(\frac{\partial l}{\partial \mathbf{Y}_k}(3\mathbf{I} - \mathbf{Y}_{k-1}\mathbf{Z}_{k-1}) - \mathbf{Z}_{k-1}\frac{\partial l}{\partial \mathbf{Z}_k}\mathbf{Z}_{k-1} - \mathbf{Z}_{k-1}\mathbf{Y}_{k-1}\frac{\partial l}{\partial \mathbf{Y}_k}\right)$$

$$\frac{\partial l}{\partial \mathbf{Z}_{k-1}} = \frac{1}{2}\left((3\mathbf{I} - \mathbf{Y}_{k-1}\mathbf{Z}_{k-1})\frac{\partial l}{\partial \mathbf{Z}_k} - \mathbf{Y}_{k-1}\frac{\partial l}{\partial \mathbf{Y}_k}\mathbf{Y}_{k-1} - \frac{\partial l}{\partial \mathbf{Z}_k}\mathbf{Z}_{k-1}\mathbf{Y}_{k-1}\right)$$

[iSQRT-COV] Peihua Li, Jiangtao Xie, Qilong Wang and Zilin Gao. Towards Faster Training of Global Covariance Pooling Networks by Iterative Matrix Square Root Normalization. In *CVPR*, 2018.

Iterative Matrix Square Root Normalization (CVPR’18)



Guarantee convergence

$$\left\| \mathbf{A} - \mathbf{I} \right\| < 1$$

Gradient

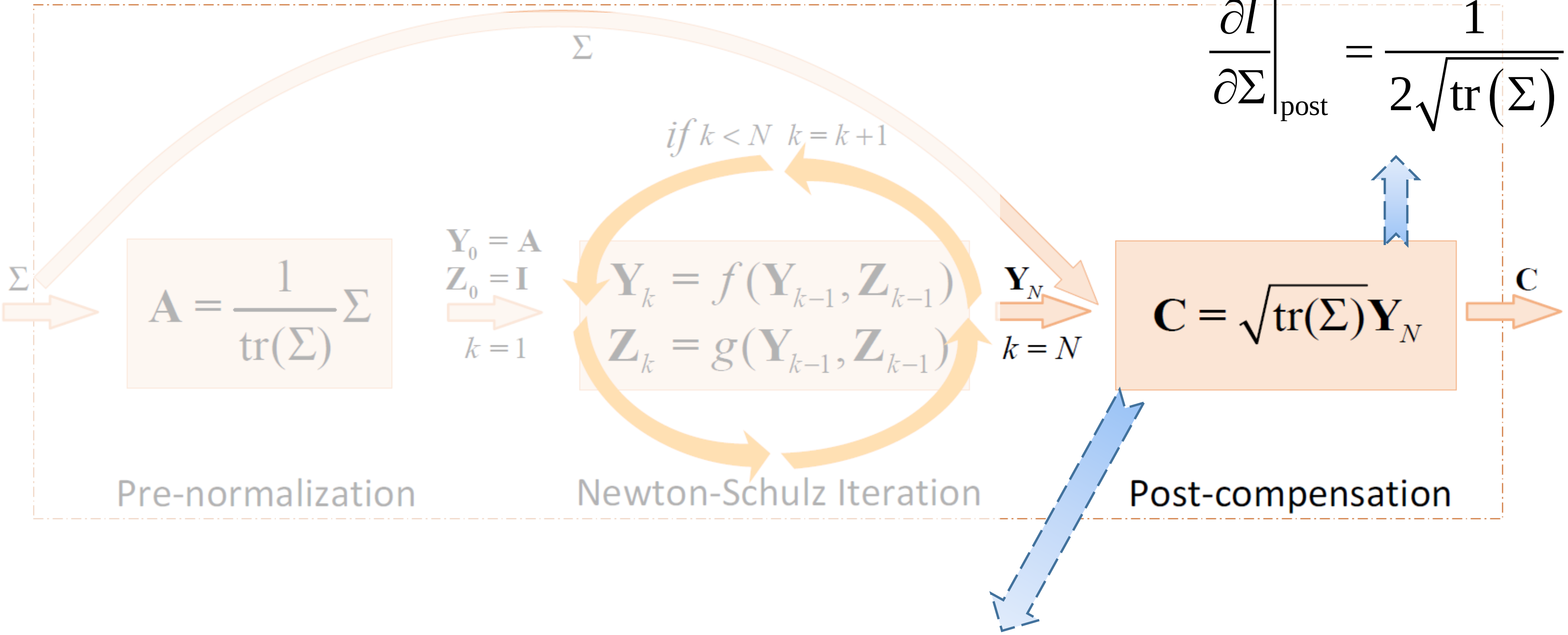
$$\frac{\partial l}{\partial \Sigma} = -\frac{1}{\left(\mathrm{tr}\left(\Sigma\right)\right)^2}\mathrm{tr}\left(\left(\frac{\partial l}{\partial \mathbf{A}}\right)^T \Sigma\right)\mathbf{I} + \frac{1}{\mathrm{tr}\left(\Sigma\right)}\frac{\partial l}{\partial \mathbf{A}} + \frac{1}{2\sqrt{\mathrm{tr}\left(\Sigma\right)}}\mathrm{tr}\left(\left(\frac{\partial l}{\partial \mathbf{C}}\right)^T \mathbf{Y}_N\right)\mathbf{I}$$

Iterative Matrix Square Root Normalization

Gradient

$$\frac{\partial l}{\partial \mathbf{Y}_N} = \sqrt{\operatorname{tr}(\Sigma)}\left(\frac{\partial l}{\partial \mathbf{C}}\right)$$

$$\left.\frac{\partial l}{\partial \Sigma}\right|_{\text{post}} = \frac{1}{2\sqrt{\operatorname{tr}(\Sigma)}}\operatorname{tr}\left(\left(\frac{\partial l}{\partial \mathbf{C}}\right)^T \mathbf{Y}_N\right)\mathbf{I}$$



Impact of post-compensation on iSQRT-COV with ResNet-50 architecture on ImageNet.

Pre-normalization	Post-compensation	Top-1 Err.	Top-5 Err.
Trace	w/o	N/A	N/A
	w/ BN [11]	23.12	6.60
	w/ Trace	**22.14**	**6.22**

Evaluation on IMAGENET

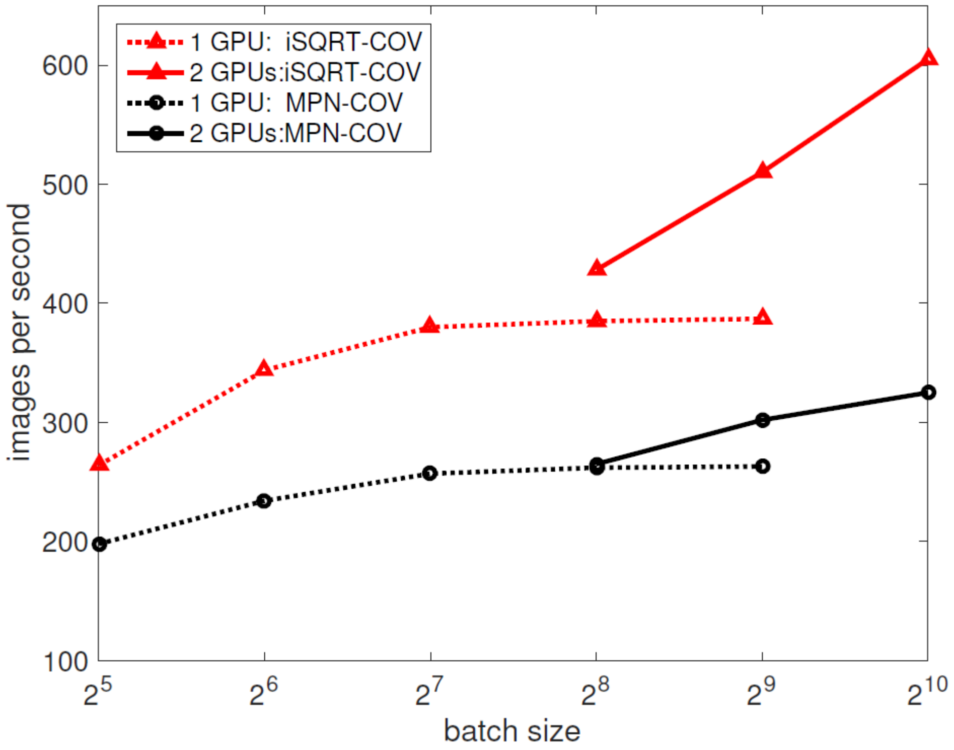
Time (FP+BP, ms) of single meta-layer with AlexNet architecture on ImageNet

Method		Language	bottleneck	Time	Memory
iSQRT-COV (N=3)		C++	N/A	**0.81 (0.26)**	0.627
iSQRT-COV (N=5)		C++	N/A	**1.41 (0.41)**	**1.129**
MPN-COV [21]		C++&M	EIG	2.58 (2.41)	0.377
Impro. B-CNN [24]	FP and BP based on SVD	M	SVD or EIG	13.51 (11.19)	0.501
	FP by NS Iter., BP by Lyap.			13.91 (2.09)	
G^2DeNet [32]		M	SVD	8.56 (4.76)	0.505

[iSQRT-COV] Peihua Li, Jiangtao Xie, Qilong Wang and Zilin Gao. Towards Faster Training of Global Covariance Pooling Networks by Iterative Matrix Square Root Normalization. In *CVPR*, 2018.

Evaluation on IMAGENET

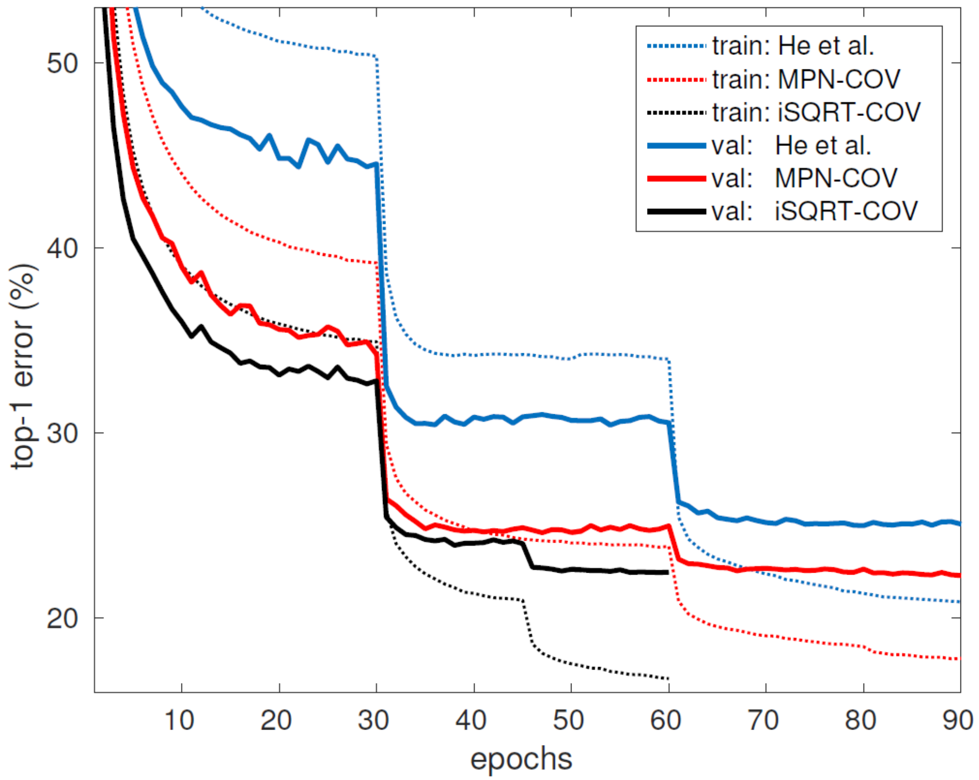
• Images per second (FP+BP) of network training with AlexNet architecture.



Our iSQRT-COV network can make better use of computing power of multi-GPU than MPN-COV

Evaluation on IMAGENET

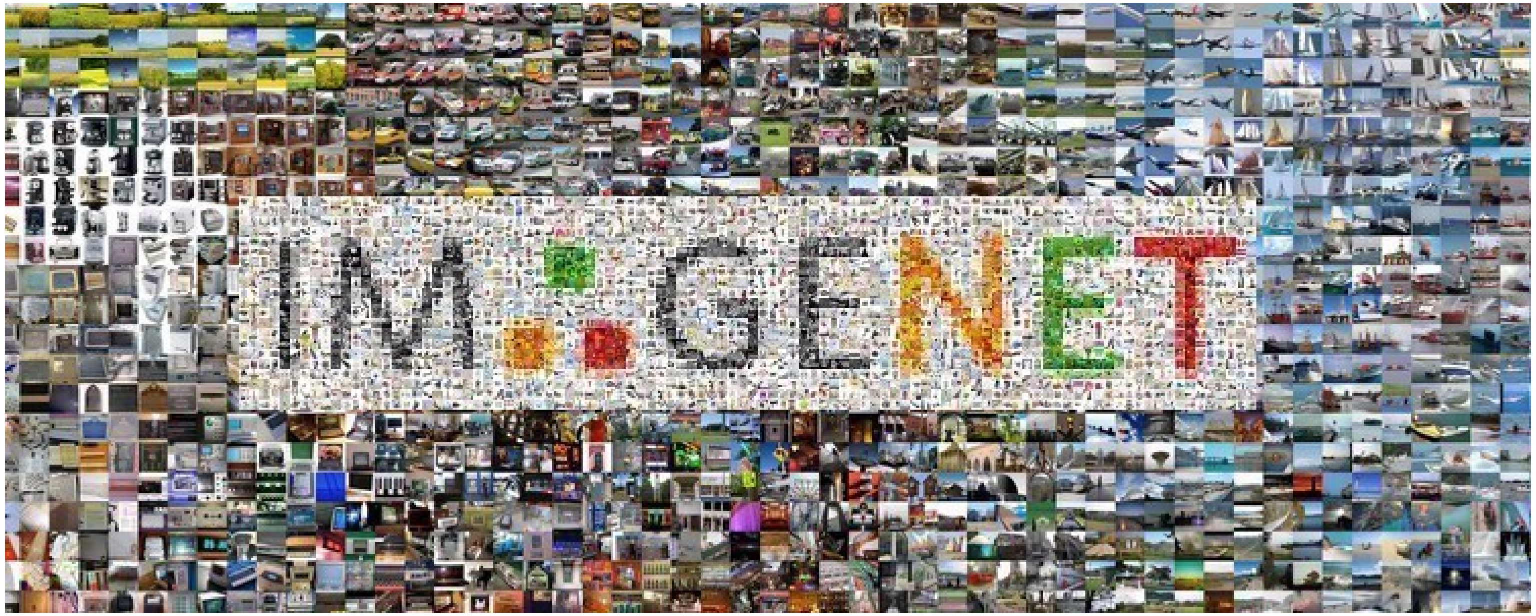
- Convergence curves of different networks trained with ResNet-50 architecture on ImageNet.



Our proposed iSQRT-COV network can converge in much less epochs.

Evaluation on IMAGENET

- Evluation on ImageNet



Classes: 1000
Train: 1.28 million
Validation: 50k
Test: 100k

http://www.image-net.org/

Evaluation on IMAGENET

Error comparison of second-order networks with first-order ones on ImagetNet

Method	Model	Top-1 Err.	Top-5 Err.
He et al. [8]		24.7	7.8
FBN [23]		24.0	7.1
SORT [35]	ResNet-50	23.82	6.72
MPN-COV [21]		22.73	6.54
iSQRT-COV		**22.14**	**6.22**
He et al. [8]	ResNet-101	23.6	7.1
iSQRT-COV		**21.21**	**5.68**
He et al. [8]	ResNet-152	23.0	6.7

[iSQRT-COV] Peihua Li, Jiangtao Xie, Qilong Wang and Zilin Gao. Towards Faster Training of Global Covariance Pooling Networks by Iterative Matrix Square Root Normalization. In *CVPR*, 2018.

Generalization to Small-scale Datasets

- Birds (CUB-200-2011)

Classes	Images
200	11,788

- Aircrafts (FGVC-aircraft)

Classes	Images
100	10,000

- Cars (Stanford cars)

Classes	Images
196	16,185

[iSQRT-COV] Peihua Li, Jiangtao Xie, Qilong Wang and Zilin Gao. Towards Faster Training of Global Covariance Pooling Networks by Iterative Matrix Square Root Normalization. In *CVPR*, 2018.

Generalization to Small-scale Datasets

Model & Method		Dim	Birds	Aircrafts	Cars
ResNet-50	MPN-COV	32K	**88.1**	**90.0**	**92.8**
	CBP[35]	14K	81.6	81.6	88.6
	KP[34]	14K	84.7	85.7	91.1
VGG-16	MPN-COV	32K	87.2	90.0	92.5
	NetVLAD[36]	32K	81.9	81.8	88.6
	CBP[35]	8K	84.3	84.1	91.2
	KP[34]	13K	86.2	86.9	92.4
	LRBP[37]	10K	84.2	87.3	90.9
	Im. B-CNN[38]	262K	85.8	88.5	92.0
	G^2DeNet[9]	263K	87.1	89.0	92.5
	HIHCA[39]	9K	85.3	88.3	91.7
MPN-COV with ResNet-101		32K	**88.7**	**91.4**	**93.3**

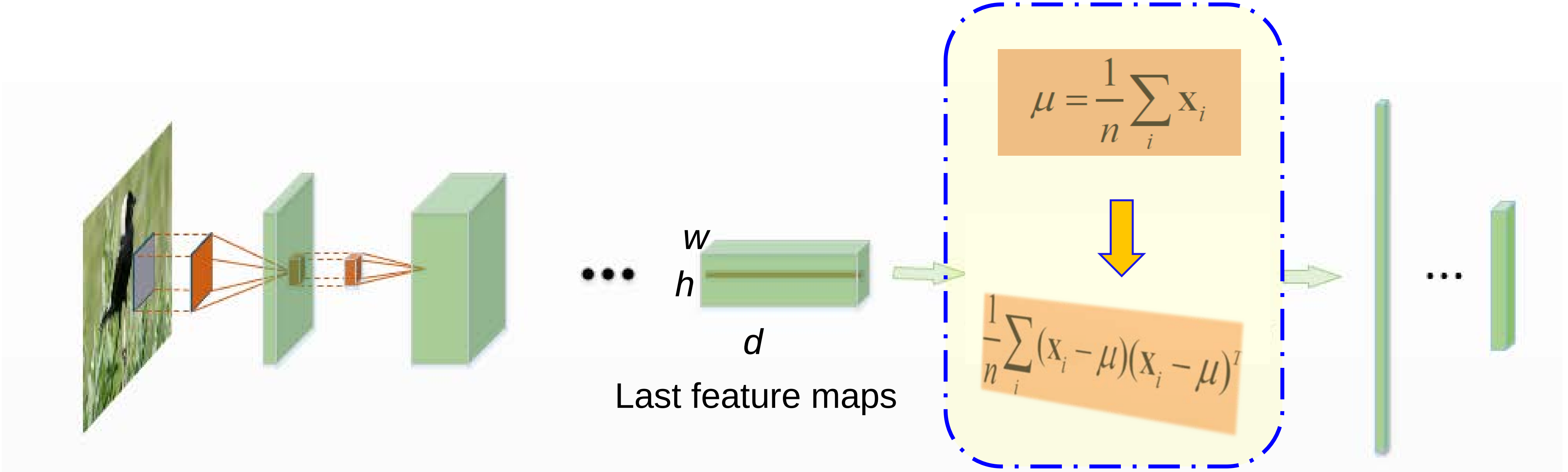
Generalization to Small-scale Datasets

Model & Method		Dim	Birds	Aircrafts	Cars
ResNet-50	MPN-COV	32K	**88.1**	**90.0**	**92.8**
	CBP[35]	14K	81.6	81.6	88.6
	KP[34]	14K	84.7	85.7	91.1
VGG-16	MPN-COV	32K	87.2	90.0	92.5
	NetVLAD[36]	32K	81.9	81.8	88.6
	CBP[35]	8K	84.3	84.1	91.2
	KP[34]	13K	86.2	86.9	92.4
	LRBP[37]	10K	84.2	87.3	90.9
	Im. B-CNN[38]	262K	85.8	88.5	92.0
	G^2DeNet[9]	263K	87.1	89.0	92.5
	HIHCA[39]	9K	85.3	88.3	91.7
MPN-COV with ResNet-101		32K	**88.7**	**91.4**	**93.3**

Generalization to Small-scale Datasets

Model & Method		Dim	Birds	Aircrafts	Cars
ResNet-50	MPN-COV	32K	**88.1**	**90.0**	**92.8**
	CBP[35]	14K	81.6	81.6	88.6
	KP[34]	14K	84.7	85.7	91.1
VGG-16	MPN-COV	32K	87.2	90.0	92.5
	NetVLAD[36]	32K	81.9	81.8	88.6
	CBP[35]	8K	84.3	84.1	91.2
	KP[34]	13K	86.2	86.9	92.4
	LRBP[37]	10K	84.2	87.3	90.9
	Im. B-CNN[38]	262K	85.8	88.5	92.0
	G^2DeNet[9]	263K	87.1	89.0	92.5
	HIHCA[39]	9K	85.3	88.3	91.7
MPN-COV with ResNet-101		32K	**88.7**	**91.4**	**93.3**

Why not a probability distribution?



Image

Local Representation of *image regions* of increasing receptive field

Global Representation of *image*

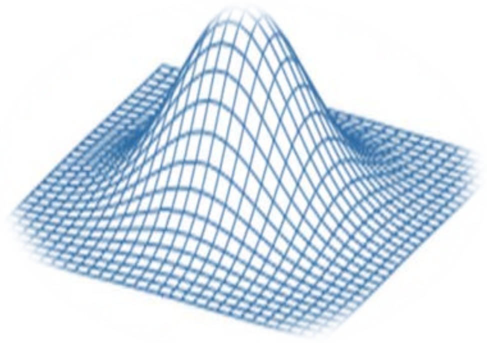
Classifier

Outline

- Global Covariance Pooling
- Matrix Power Normalizatin and Fast Training
- **Global Distribution Modeling for CNN**
- Conclusion

G^2DeNet (CVPR’17)

- *Why not a probability distribution*?



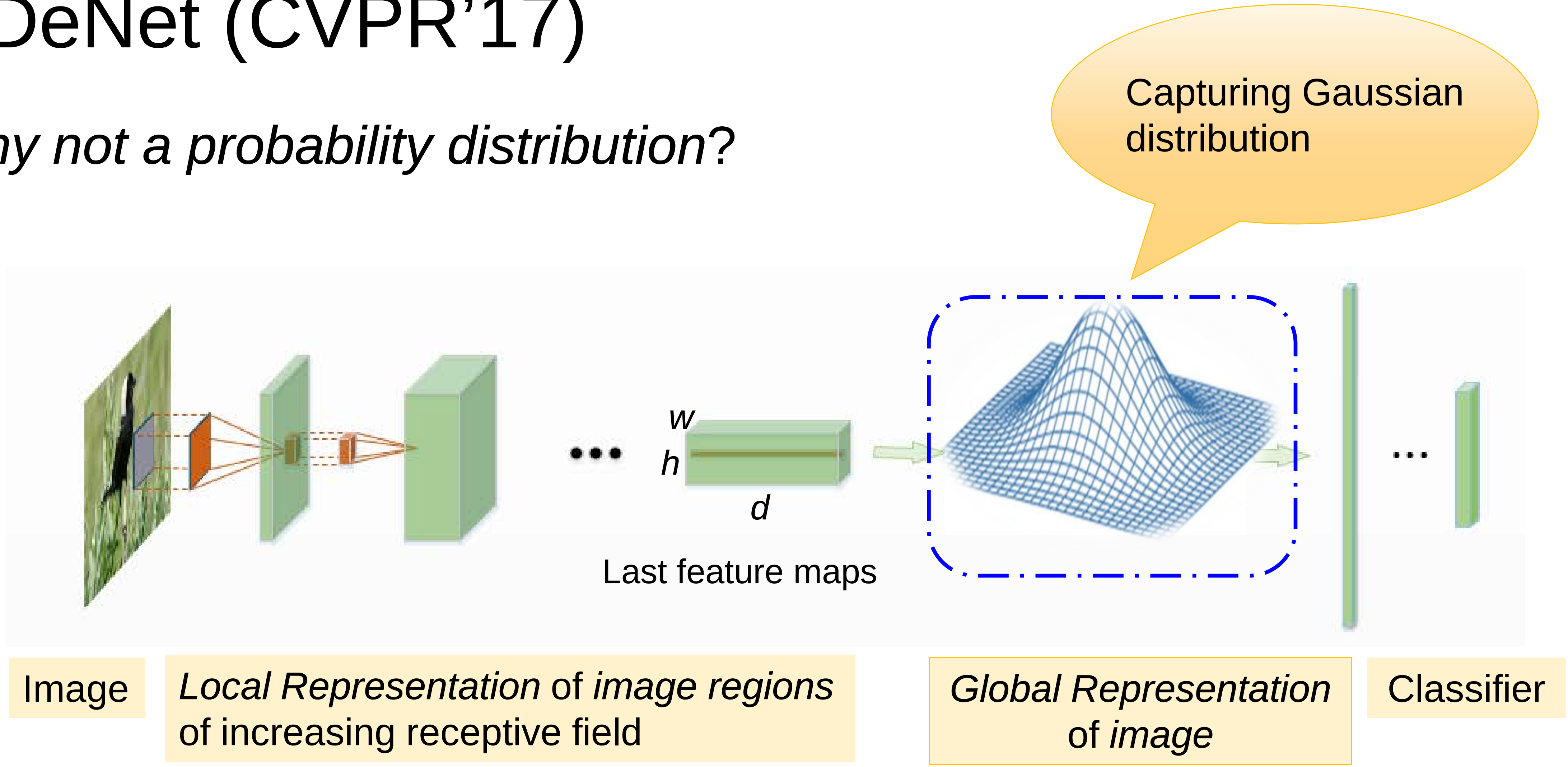
- ➢ **Maximum entropy distribution** with specified mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$
  - – Least prior information
  - – Physical systems tend to be of maximum entropy

$$p(\mathbf{x}) = \frac{1}{(2\pi)^{\frac{d}{2}}|\boldsymbol{\Sigma}|^{\frac{1}{2}}} \exp\Big(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T\boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\Big)$$

[G^2DeNet] Qilong Wang, Peihua Li, Lei Zhang. G^2DeNet: Global Gaussian Distribution Embedding Network and Its Application to Visual Recognition. In *CVPR*, 2017 (*Oral*).

G^{2}DeNet (CVPR'17)

- *Why not a probability distribution?*

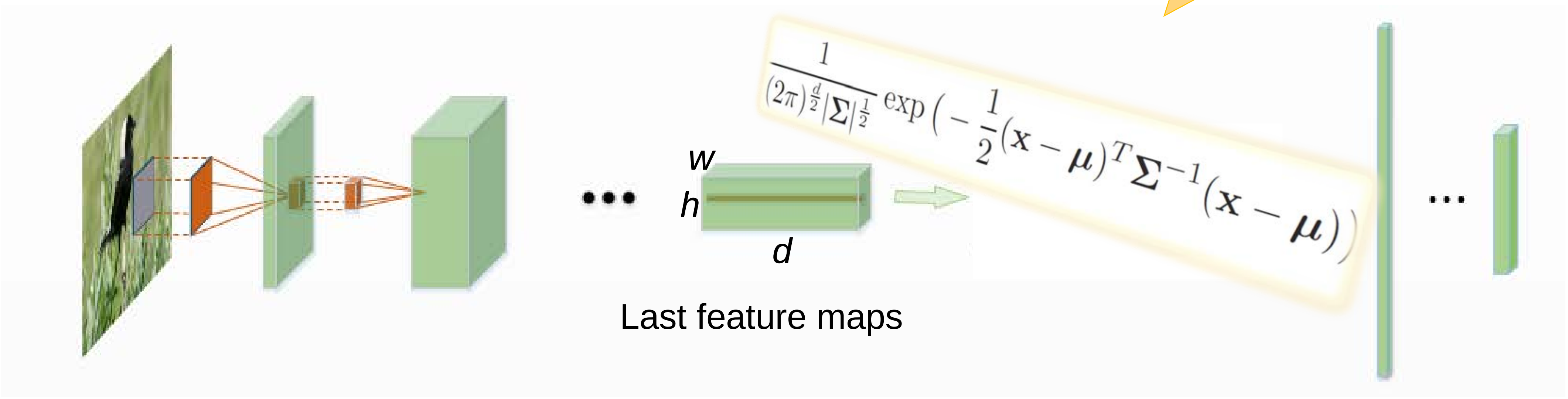


[G^{2}DeNet] Qilong Wang, Peihua Li, Lei Zhang. G^{2}DeNet: Global Gaussian Distribution Embedding Network and Its Application to Visual Recognition. In *CVPR*, 2017 (*Oral*).

G²DeNet (CVPR'17)

- *Why not* Global Covariance + Average Pooling?

Capturing Gaussian distribution



Image

Local Representation of *image regions* of increasing receptive field

Global Representation of *image*

Classifier

[G²DeNet] Qilong Wang, Peihua Li, Lei Zhang. G²DeNet: Global Gaussian Distribution Embedding Network and Its Application to Visual Recognition. In *CVPR*, 2017 (*Oral*).

G^2DeNet (CVPR'17)

Q: How to construct our trainable global Gaussian embedding layer?

A: The key is to give the explicit forms of Gaussian distributions.

- **Forward Propagation**
  - **Riemannian Geometry Structure**
  - **Algebraic Structure**
- **Backward Propagation**
  - **Differentiable**

Gaussian embedding [TPAMI’17]

The space of Gaussians is a Riemannian manifold having special geometric structure.

[TPAMI’17] shows space of Gaussians is equipped with a Lie group structure.

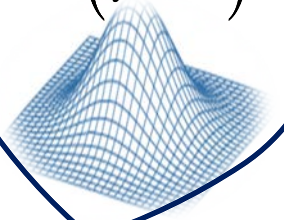
Cholesky decomp.

$$\boldsymbol{\Sigma} = \mathbf{L}^{-T}\mathbf{L}^{-1}$$

left polar decomp.

$$\mathbf{A}_{\boldsymbol{\mu},\mathbf{L}^{-T}} \to \mathbf{YO}$$

$$\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$$



Gaussian

$$\mathbf{A}_{\boldsymbol{\mu},\mathbf{L}^{-T}} = \begin{bmatrix} \mathbf{L}^{-T} & \boldsymbol{\mu} \\ \mathbf{0}^T & 1 \end{bmatrix}$$

Positive upper triangular matrix

$$\mathbf{Y} = \begin{bmatrix} \boldsymbol{\Sigma} + \boldsymbol{\mu}\boldsymbol{\mu}^T & \boldsymbol{\mu} \\ \boldsymbol{\mu}^T & 1 \end{bmatrix}^{\frac{1}{2}}$$

Symmetric positive definite (SPD) matrix

[TPAMI’17] Peihua Li, Qilong Wang, Hui Zeng and Lei Zhang. Local Log-Euclidean Multivariate Gaussian Descriptor and Its Application to Image Classification. TPAMI, 2017.

Gaussian embedding [TPAMI'17]

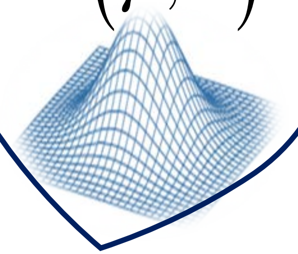
The space of Gaussians is a Riemannian manifold having special geometric structure.

[TPAMI'17] shows space of Gaussians is equipped with a Lie group structure.

Cholesky decomp.

$$\boldsymbol{\Sigma} = \mathbf{L}^{-T}\mathbf{L}^{-1}$$

$$\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$$



Gaussian

$$\mathbf{A}_{\boldsymbol{\mu},\mathbf{L}^{-T}} = \begin{bmatrix} \mathbf{L}^{-T} & \boldsymbol{\mu} \\ \mathbf{0}^{T} & 1 \end{bmatrix}$$

Positive upper triangular matrix

left polar decomp.

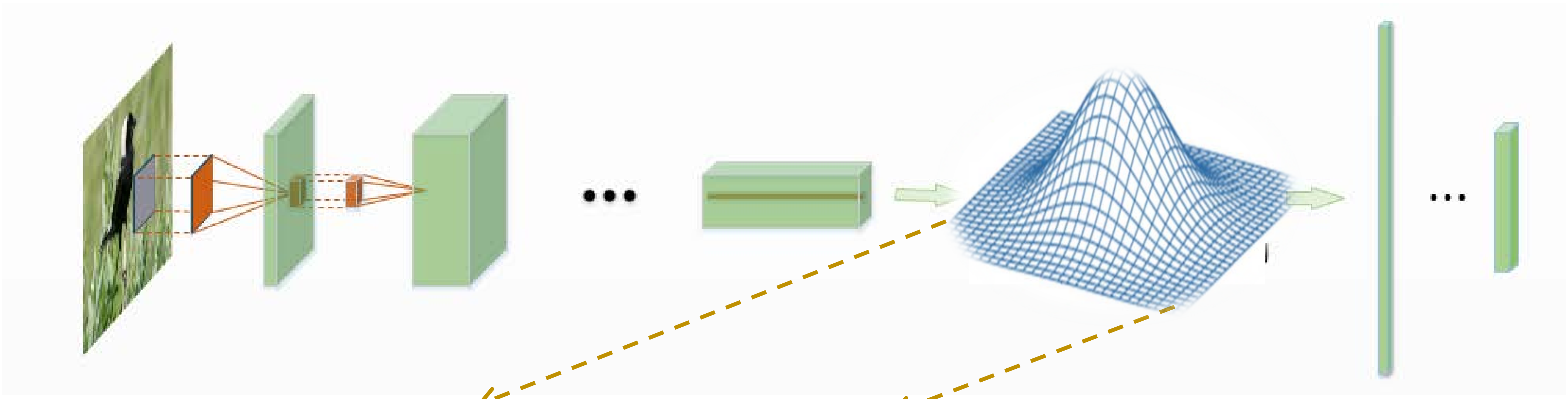
$$\mathbf{A}_{\boldsymbol{\mu},\mathbf{L}^{-T}} \rightarrow \mathbf{YO}$$

$$\mathbf{Y} = \begin{bmatrix} \boldsymbol{\Sigma} + \boldsymbol{\mu}\boldsymbol{\mu}^{T} & \boldsymbol{\mu} \\ \boldsymbol{\mu}^{T} & 1 \end{bmatrix}^{\frac{1}{2}}$$

Symmetric positive definite (SPD) matrix

[TPAMI'17] Peihua Li, Qilong Wang, Hui Zeng and Lei Zhang. Local Log-Euclidean Multivariate Gaussian Descriptor and Its Application to Image Classification. TPAMI, 2017.

G^{2}DeNet (CVPR'17)



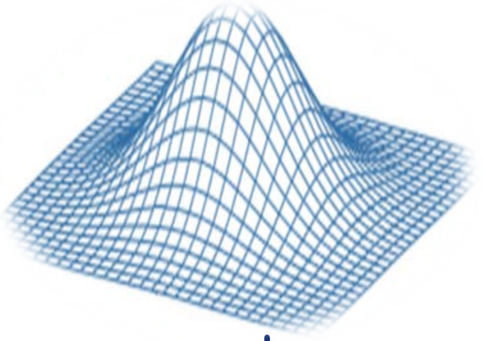
$$\begin{bmatrix} \boldsymbol{\Sigma} + \boldsymbol{\mu}\boldsymbol{\mu}^T & \boldsymbol{\mu} \\ \boldsymbol{\mu}^T & 1 \end{bmatrix}^{\frac{1}{2}}$$

[G^{2}DeNet] Qilong Wang, Peihua Li, Lei Zhang. G^{2}DeNet: Global Gaussian Distribution Embedding Network and Its Application to Visual Recognition. In *CVPR*, 2017 (*Oral*).

[TPAMI'17] Peihua Li, Qilong Wang, Hui Zeng and Lei Zhang. Local Log-Euclidean Multivariate Gaussian Descriptor and Its Application to Image Classification. TPAMI, 2017.

G^2DeNet (CVPR'17)

Gaussian Embedding :



$$\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma}) \rightarrow \mathbf{Y} = \begin{bmatrix} \boldsymbol{\Sigma} + \boldsymbol{\mu}\boldsymbol{\mu}^T & \boldsymbol{\mu} \\ \boldsymbol{\mu}^T & 1 \end{bmatrix}^{\frac{1}{2}}$$

1. Matrix Partition Sub-layer :

$$\mathbf{Y} = f_{MPL}(\mathbf{X}) = \begin{bmatrix} \boldsymbol{\Sigma} + \boldsymbol{\mu}\boldsymbol{\mu}^T & \boldsymbol{\mu} \\ \boldsymbol{\mu}^T & 1 \end{bmatrix} = \frac{1}{N}\mathbf{A}\mathbf{X}^T\mathbf{X}\mathbf{A}^T + \frac{2}{N}\left(\mathbf{A}\mathbf{X}^T\mathbf{1}\mathbf{b}^T\right)_{sym} + \mathbf{B}$$

Y is a function of convolutional features X.

2. Square-rooted SPD Matrix Sub-layer:

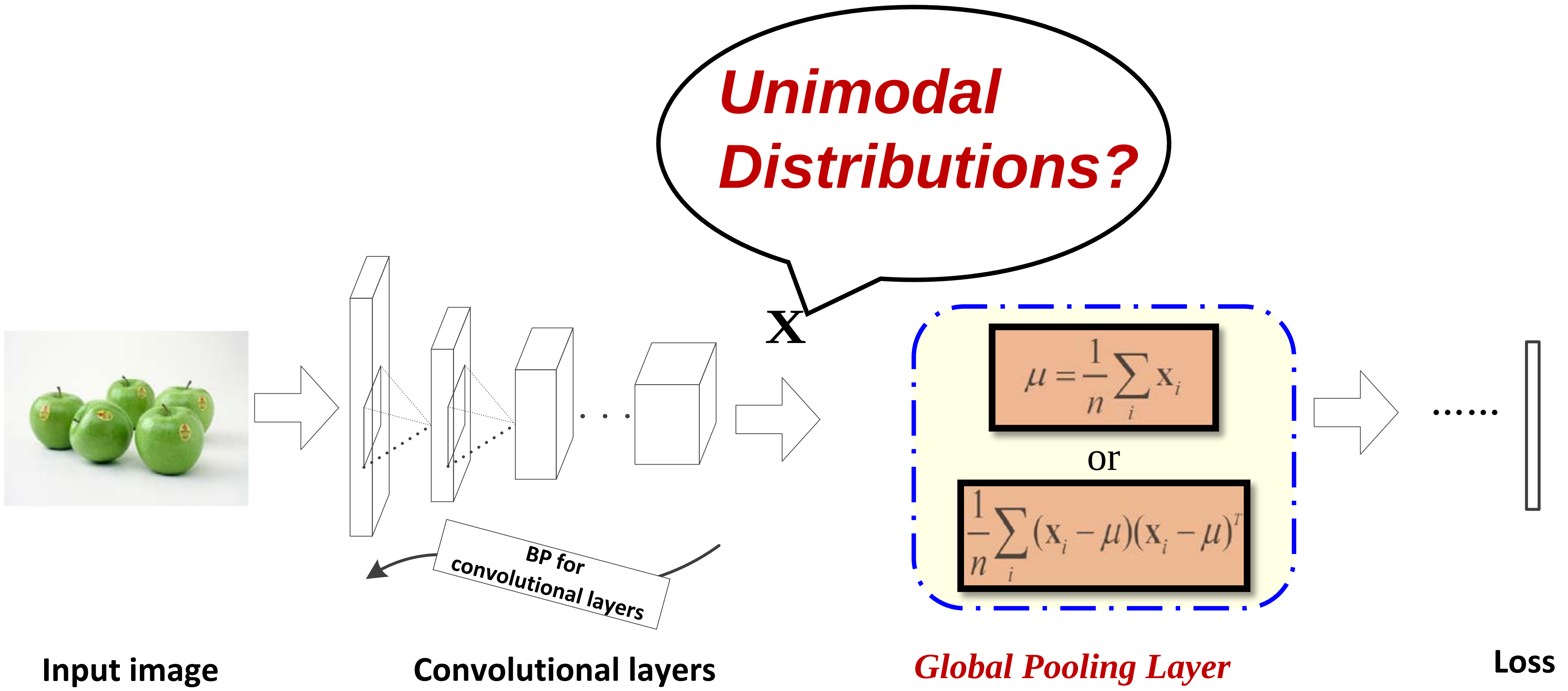
$$\mathbf{Z} = f_{ESRL}(\mathbf{Y}) = \mathbf{Y}^{\frac{1}{2}} = \mathbf{U}\boldsymbol{\Lambda}^{\frac{1}{2}}\mathbf{U}^T$$

Computing square-root of Y via SVD.

[G^2DeNet] Qilong Wang, Peihua Li, Lei Zhang. G^2DeNet: Global Gaussian Distribution Embedding Network and Its Application to Visual Recognition. In *CVPR*, 2017 (*Oral*).

[TPAMI'17] Peihua Li, Qilong Wang, Hui Zeng and Lei Zhang. Local Log-Euclidean Multivariate Gaussian Descriptor and Its Application to Image Classification. TPAMI, 2017.

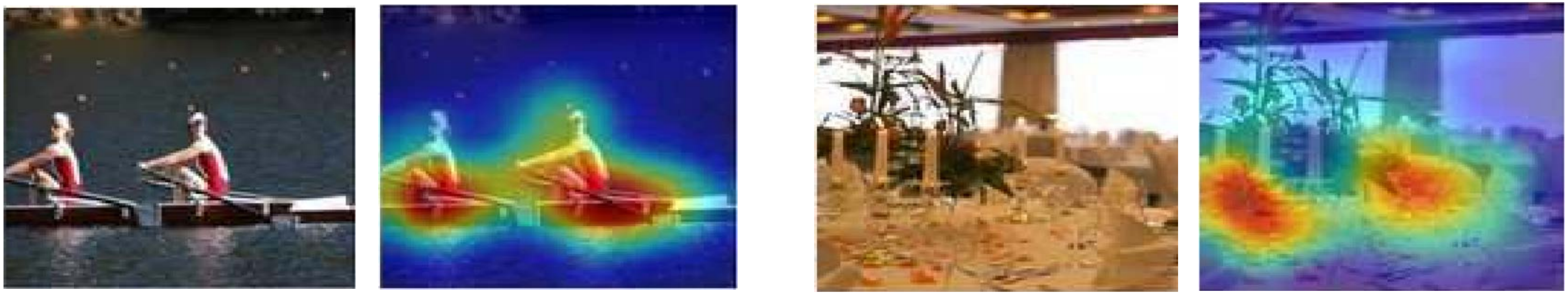
Existing Deep CNNs



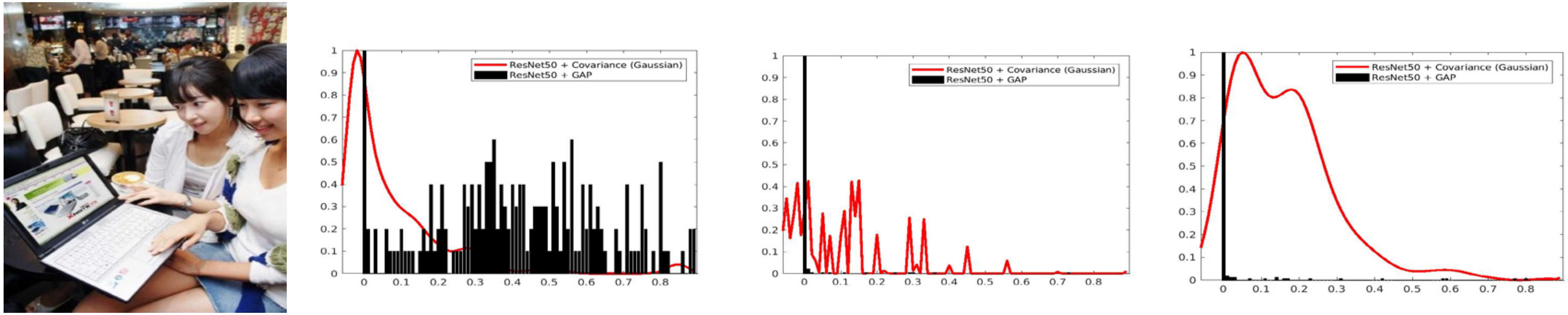
Existing global pooling layers assume unimodal distributions, which cannot fully capture statistics of convolutional activations.

Qilong Wang *et al*. Global Gated Mixture of Second-order Pooling for Improving Deep Convolutional Neural Networks. In NIPS, 2018.

Existing Deep CNNs



Learning Deep Features for Discriminative Localization. CVPR, 2016



Qilong Wang *et al*. Global Gated Mixture of Second-order Pooling for Improving Deep Convolutional Neural Networks. In NIPS, 2018.

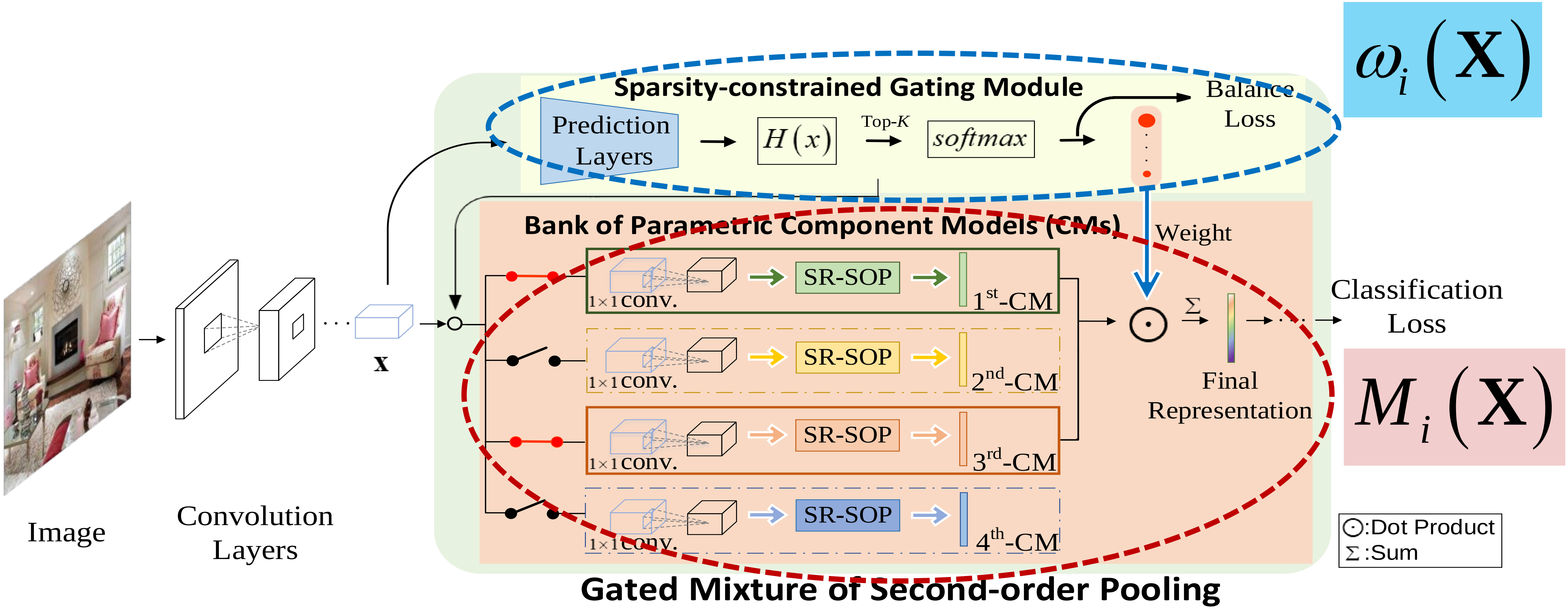
Mixture Model

- Ensemble of multiple models
  - High computational cost as number of CMs gets large, while a small number of CMs may be insufficient for characterizing complex distributions.
  - Simple direct ensemble will make all CMs tend to learn similar characteristics.

$$\boxed{\mathbf{y} = \sum_{i=1}^{N} \omega_i(\mathbf{X}) M_i(\mathbf{X}), \;\; s.t., \;\; \sum_{i=1}^{N} \omega_i(\mathbf{X}) = 1,}$$

Qilong Wang *et al*. Global Gated Mixture of Second-order Pooling for Improving Deep Convolutional Neural Networks. In NIPS, 2018.

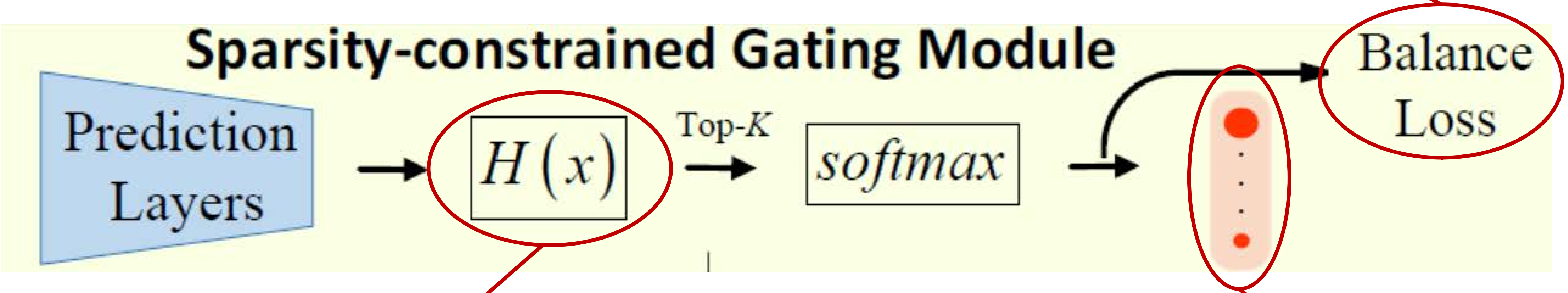
Deep CNNs with GM-SOP



Qilong Wang *et al*. Global Gated Mixture of Second-order Pooling for Improving Deep Convolutional Neural Networks. In NIPS, 2018.

Sparsity-constrained Gating Module

$$L_B = \alpha \left(\frac{std(\sum_{s=1}^{S} \boldsymbol{\omega}(\mathbf{X}_s))}{\mu(\sum_{s=1}^{S} \boldsymbol{\omega}(\mathbf{X}_s))} \right)^2,$$



$$H_i(f(\theta_g; \mathbf{X})) = \mathbf{W}_i^g f(\theta_g; \mathbf{X}) + \gamma \cdot \log(1 + \exp(\mathbf{W}_i^n f(\theta_g; \mathbf{X}))).$$

$$\omega_i(\mathbf{X}) = \frac{\exp(\text{Top-}K(H_i(f(\theta_g; \mathbf{X}))))}{\sum_{i=1}^{N} \exp(\text{Top-}K(H_i(f(\theta_g; \mathbf{X}))))}.$$

Qilong Wang *et al*. Global Gated Mixture of Second-order Pooling for Improving Deep Convolutional Neural Networks. In NIPS, 2018.

Components Models

$$\mathbf{y} = \sum_{i=1}^{N} \omega_i(\mathbf{X}) \boxed{M_i(\mathbf{X})}, \quad s.t., \sum_{i=1}^{N} \omega_i(\mathbf{X}) = 1,$$

Method	Top-1 Error	Top-5 Error
AlexNet [18]	41.8	19.2
MPN-COV [21]	38.51	17.60
B-CNN [25]	39.89	18.32
DeepO$_2$P [12]	42.16	19.62
Impro. B-CNN* [24]	40.75	18.91
G^{2}DeNet [32]	38.71	17.66
iSQRT-COV(Frob.)	38.78	17.67
iSQRT-COV(trace)	**38.45**	**17.52**

- SR-SOP seems to be a good choice for CM

Qilong Wang *et al*. Global Gated Mixture of Second-order Pooling for Improving Deep Convolutional Neural Networks. In NIPS, 2018.

Parametric Components Models

SR-SOP **(Existing Top Unimodal Pooling)** [CVPR2017, ICCV2017, CVPR 2018]

$$\mathbf{Z} = \boldsymbol{\Sigma}^{\frac{1}{2}} \qquad \boldsymbol{\Sigma} = \mathbf{X}^{\mathrm{T}}\hat{\mathbf{J}}\mathbf{X} \qquad \hat{\mathbf{J}}$$

~ *constant*

~ *normal Gaussian distribution*

Parametric SR-SOP (Ours)

$$\mathbf{Z} = \hat{\boldsymbol{\Sigma}}^{\frac{1}{2}}, \quad \hat{\boldsymbol{\Sigma}} = \left(\mathbf{L}_J \cdots \mathbf{L}_1\mathbf{X}\right)^{\mathrm{T}}\left(\mathbf{L}_J \cdots \mathbf{L}_1\mathbf{X}\right) \qquad \mathbf{G}_j = \mathbf{L}_j^{T}\mathbf{L}_j$$

~ *trainable*

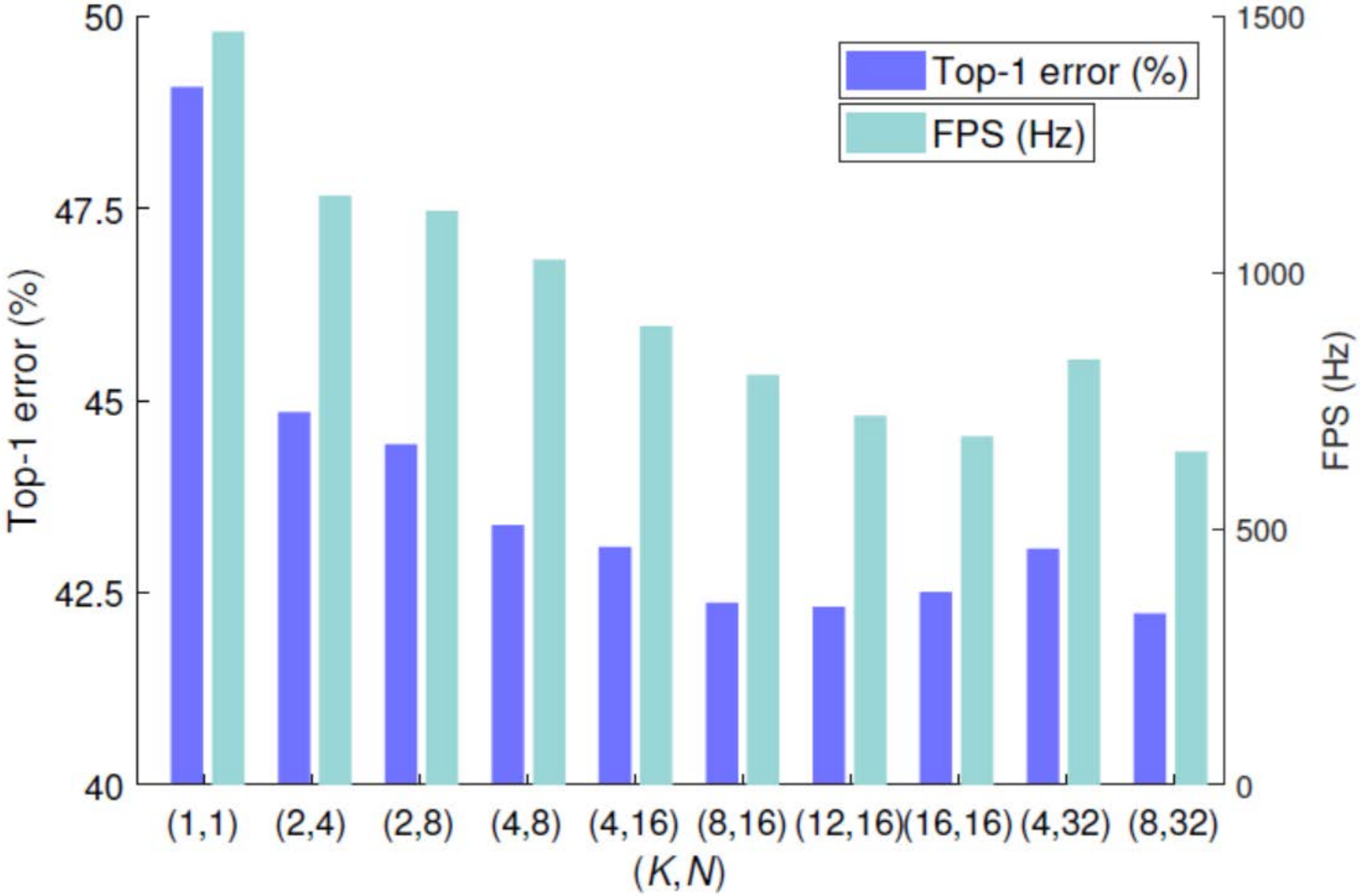
~ *multivariate generalized Gaussian distribution*

Qilong Wang *et al*. Global Gated Mixture of Second-order Pooling for Improving Deep Convolutional Neural Networks. In NIPS, 2018.

Ablation study

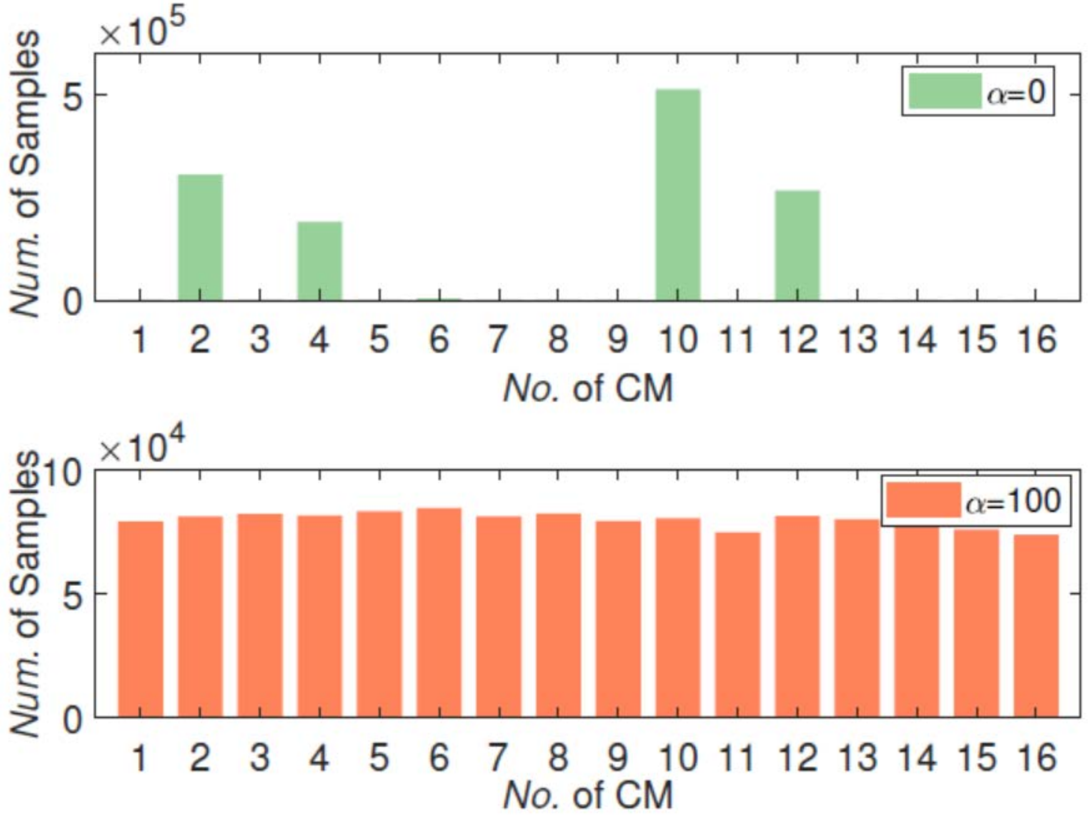
Training:
1.28M Images

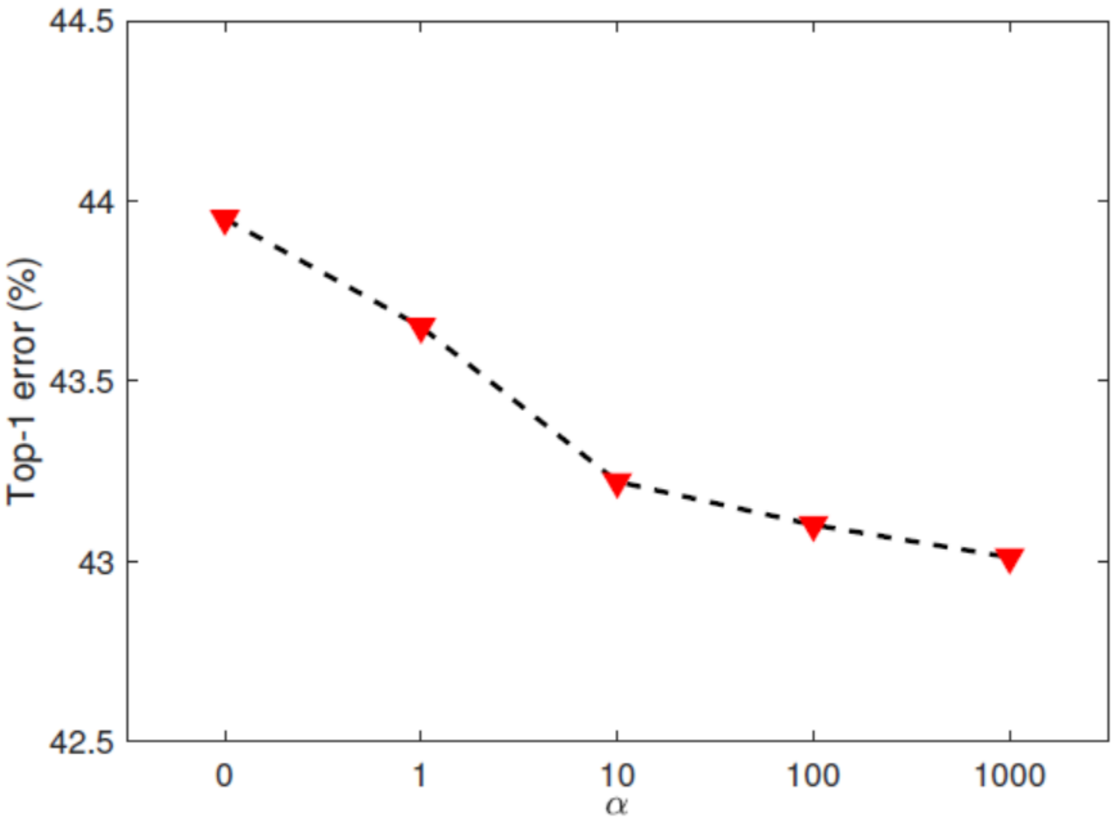
Testing:
50K Images



Numbers of N and K

Ablation study





Effect of Parameter α

Experiments on ImageNet-1K

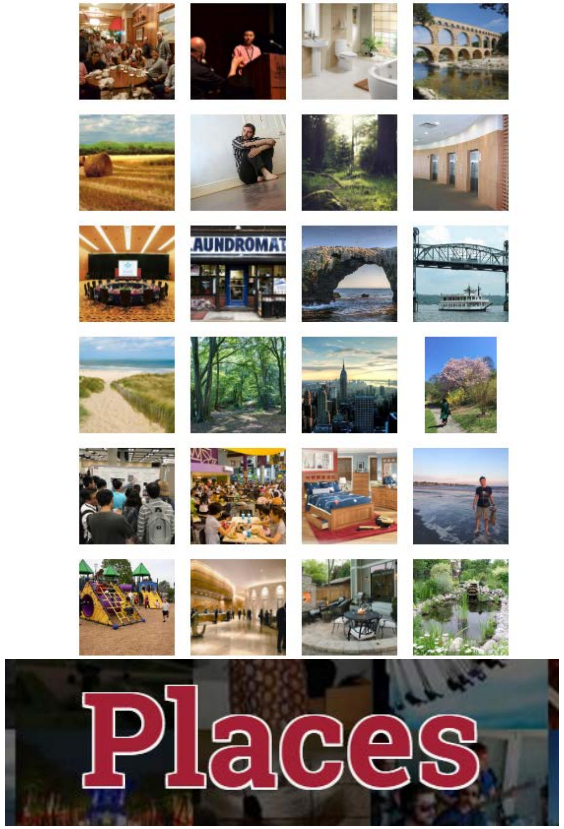
Training: 1.28M

Testing: 50K

Methods	Backbone Models	Top-1 Error	Top-5 Error	Top-1 Increment	Top-5 Increment
GAP	ResNet-18	49.08	24.25	+0.00	+0.00
SOP		40.32	18.68	+8.76	+5.57
GM-SOP		38.21	17.01	**+10.87**	**+7.24**
GAP	ResNet-50	41.42	18.14	**+5.69**	**+3.18**
GM-SOP		35.73	14.96		
GAP	WRN-36-2	39.55	16.57	**+7.22**	**+4.22**
GM-SOP		32.33	12.35		

Qilong Wang *et al*. Global Gated Mixture of Second-order Pooling for Improving Deep Convolutional Neural Networks. In NIPS, 2018.

Experiments on Places365



Training:
~1.8M

Testing:
36.5K

Methods	Backbone Models	Top-1 Error	Top-5 Error
GAP	ResNet-18	49.96	19.19
GM-GAP		**48.07**	**17.84**
GAP-8256d		49.99	19.32
SOP		48.11	18.01
SR-SOP		47.48	17.52
GM-SOP		**47.18**	**17.02**

Qilong Wang *et al*. Global Gated Mixture of Second-order Pooling for Improving Deep Convolutional Neural Networks. In NIPS, 2018.

Deep BoVW - NetVLAD

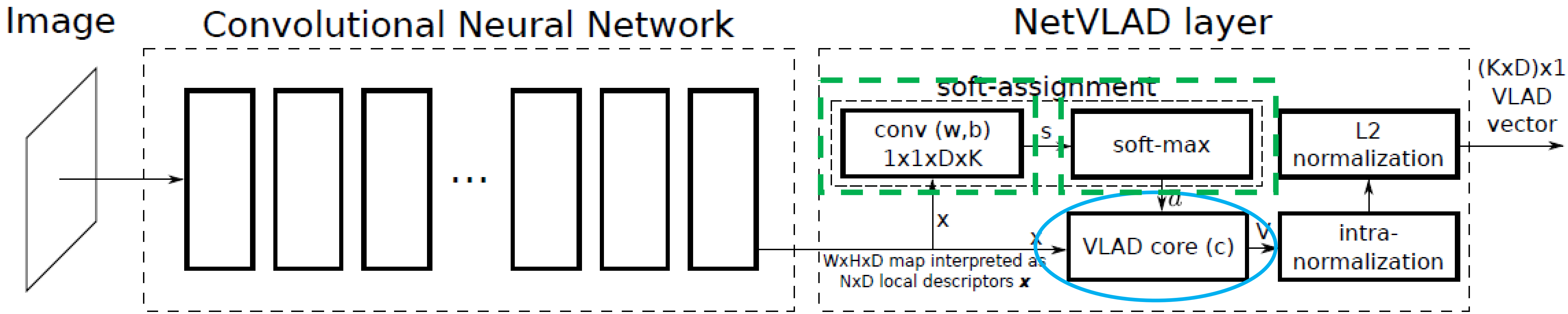


Figure 2. **CNN architecture with the NetVLAD layer.** The layer can be implemented using standard CNN layers (convolutions, softmax, L2-normalization) and one easy-to-implement aggregation layer to perform aggregation in equation (4) ("VLAD core"), joined up in a directed acyclic graph. Parameters are shown in brackets.

$$\bar{a}_k(\mathbf{x}_i) = \frac{e^{-\alpha\|\mathbf{x}_i - \mathbf{c}_k\|^2}}{\sum_{k'} e^{-\alpha\|\mathbf{x}_i - \mathbf{c}_{k'}\|^2}},$$

$$\mathbf{w}_k = 2\alpha\mathbf{c}_k$$

$$b_k = -\alpha\|\mathbf{c}_k\|^2$$

$$\bar{a}_k(\mathbf{x}_i) = \frac{e^{\mathbf{w}_k^T\mathbf{x}_i + b_k}}{\sum_{k'} e^{\mathbf{w}_{k'}^T\mathbf{x}_i + b_{k'}}},$$

$$V(j,k) = \sum_{i=1}^{N} a_k(\mathbf{x}_i)\,(x_i(j) - c_k(j))$$

$$V(j,k) = \sum_{i=1}^{N} \frac{e^{\boxed{\mathbf{w}_k^T\mathbf{x}_i + b_k}}}{\sum_{k'} e^{\mathbf{w}_{k'}^T\mathbf{x}_i + b_{k'}}}(x_i(j) - c_k(j))$$

$$\sigma(\mathbf{z})_j = \frac{e^{z_j}}{\sum_{k=1}^{K} e^{z_k}} \quad \text{for } j = 1, \ldots, K.$$

Relja Arandjelovi´c, et al. NetVLAD: CNN architecture for weakly supervised place recognition. CVPR, 2016.

Deep BoVW – Simple NetFV

NetVLAD: $\boxed{\mathbf{x} - \boldsymbol{\mu}}$

Simple NetFV: $\boxed{[\mathbf{x} - \boldsymbol{\mu}, (\mathbf{x} - \boldsymbol{\mu}) \odot (\mathbf{x} - \boldsymbol{\mu})]}$

Lin et al. Bilinear CNNs for Fine-grained Visual Recognition. TPAMI, 2017.

Comparison

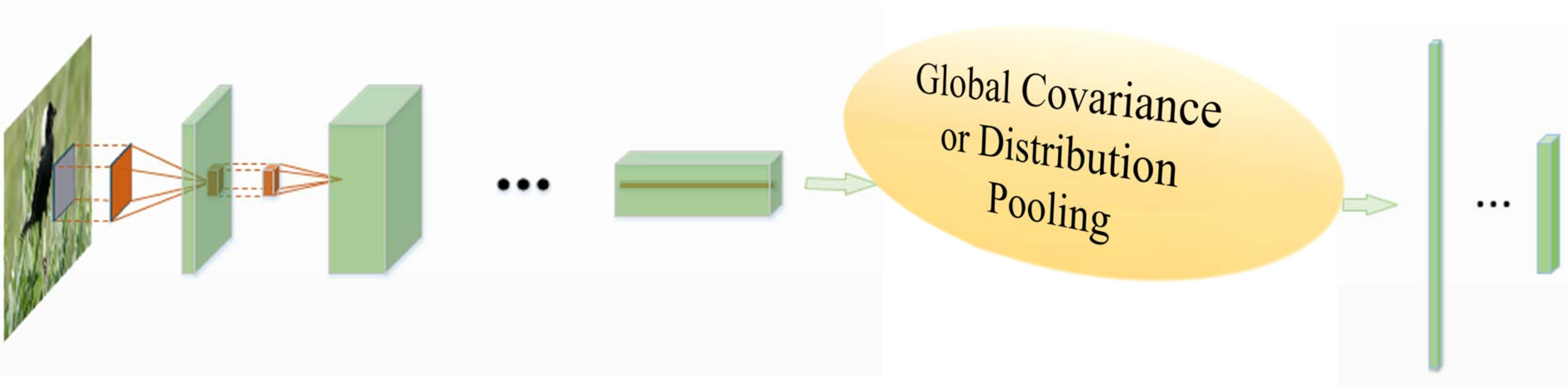
Methods	Birds CUB-200-2011	FGVC-Aircraft	FGVC-Cars
NetFV [TPAMI’17]	79.9	79.0	86.2
NetVLAD [CVPR’16]	81.9	81.8	88.6
B-CNN [ICCV’15]	84.1	84.1	91.3
G^{2}DeNet (Ours)	**87.1**	**89.0**	**92.5**

FGVC

Methods	Backbone	Top-1 error	Top-5 error
NetVLAD [CVPR’16]	ResNet-18	45.16	21.73
SOP		40.32	18.68
GM-SOP		**38.21**	**17.01**

64x64 ImageNet-1K

Conclusion



- ✓ Performing and generalizing much better
- ✓ Statistical and geometrical insights
- ✓ Fast convergence, computation-efficient

Related Publications

Global Covariance Pooling

[1] Peihua Li, Jiangtao Xie, Qilong Wang and Wangmeng Zuo. Is Second-order Information Helpful for Large-scale Visual Recognition? In *ICCV*, 2017.

Global Gaussian Pooling and Gaussian Embedding

[2] Qilong Wang, Peihua Li, Lei Zhang. G^2DeNet: Global Gaussian Distribution Embedding Network and Its Application to Visual Recognition. In *CVPR*, 2017 (*Oral*).

[3] Peihua Li, Qilong Wang, Hui Zeng, Lei Zhang. Local Log-Euclidean Multivariate Gaussian Descriptor and Its Application to Image Classification. *IEEE TPAMI*, 2017.

Fast Training Algorithm

[4] Peihua Li, Jiangtao Xie, Qilong Wang and Zilin Gao. Towards Faster Training of Global Covariance Pooling Networks by Iterative Matrix Square Root Normalization. In *CVPR*, 2018.

Robust Covariance Estimation

[5] Qilong Wang, Peihua Li, Wangmeng Zuo, Lei Zhang. RAID-G: Robust Estimation of Approximate Infinite Dimensional Gaussian with Application to Materiel Recognition. In *CVPR*, 2016.

Multimodal Distribution

[6] Qilong Wang, Zilin Gao, Jiangtao Xie, Wangmeng Zuo, Peihua Li. Global Gated Mixture of Second-order Pooling for Improving Deep Convolutional Neural Networks. In NIPS, 2018.

All code are (or will be) released at
http://peihuali.org

Thank you!